DRAGON USER

The independent Dragon magazine

August 1988

Contents

Letters 2
Alterations to the DOSs ... excellent service from Dragon suppliers ... *Basic09* tweak ... RAE offered ... devious disc drives.

News 4
New software from Dragonfire, Pulser, Orange ... H.C. Andersen price reductions ... advice for reviewers

Dragonsoft 6,11,19
Mandragore from Kouga, *Destiny* trio from Pulser, *Utopia* from Pulser, *Music Maker* from John Penn.

Dragonsword! 7
Paul Grade's mighty pen takes a quick stab at anything which drops its guard for second.

Paging Sam 8
P D Smith consults Motorola and comes up with more stories of the sam chip.

Crossword 10
The ninth Dragon crossword, with tapes to win.

Hard discs 12
Roger Merrick relates his experience of driving a hard disc and OS-9 on his Dragon.

Invoicer 14
Craig Anderson, like a true Dragon, knows the value of every item in his hoard.

Cartesian graphs 20
Faster mathematical drawing by R L Moss and his Dragon.

Winners and losers 22
Gordon Lee looks at some of the entries to the May competition, with the winning listings.

Write: ADVENTURE 23
Pete Gerrard looks at the ideas, the characters, and wonders what they will do next.

Adventure Trail 24
Professor Deadrock, the Japanese adventurer, sets out on another solution.

Competition 26
Another Gordon Lee brain-twister, with Preston's *Hotel on Mayfair* as the prize.

Dragon Answers 28
Brian Cadge answers some more technical queries on the inside of the Dragon.

Editorial

HERE we are, just got the new office sorted out and holiday time bearing down on me like a ten ton truck — literally, indeed, as I am moving house at the end of the week. However, thanks to Artset and the contribs it shouldn't affect this a(A)ugust magazine. Fingers crossed, toes crossed.

I need a holiday.

By the way, owing to a paper avalanche at Sunshine a few weeks ago, some people who should have done, won't have had their June renewal invoices yet. We have only just found out, and we're working on it.

By coincidence, Roger Merrick's report on his experience with Gordon Twist's hard disc system arrived in time to prove what Paul Grade is saying in his new column. The power is there for the Dragon if you look for it.

Problems with hard disc fragility are universal and not confined to the Dragon. Ask anybody with a hard disc, and watch their faces!

And now pundits predict that digital tape recording will lead data storage back to cassettes in the longer term.

Newsdesk contains an editorial rant for reviewers, in reply to those who want the secrets of fame and fortune.

And the Golden Rule? Meet the deadline — bye!

Telephone number
(01) 570 8335

Editor
HELEN ARMSTRONG

Production Editor
HELEN ARMSTRONG/ARTSET

Administration/Advertising
BOB HARRIS

Publisher
DRAGON PUBLICATIONS

Subscriptions
UK £14 for 12 issues
Overseas (surface) £20 for 12 issues

ISSN 0265-177

Address: Dragon Publications, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP, United Kingdom.

Published by Dragon Publications 1988

Typeset by Artset Limited, London NW1

Printed by Headley Brothers Ltd. Ashford, Kent

Registered at the Post Office as a newspaper. Dragon and its logo are trademarks of Eurohard Ltd.

How to submit articles

The quality of the material we can publish in *Dragon User* each month will, to a very great extent depend on the quality of the discoveries that you can make with your Dragon. The Dragon computer was launched on to the market with a powerful version of Basic, but with very poor documentation.

Articles which are submitted to *Dragon User* for publication should not be more than 3000 words long. All submissions should be typed. Please leave wide margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submitted article or program, so please keep a copy. If you want to have your program returned you must include a stamped addressed envelope.

Letters

This is your chance to air your views — send your tips, compliments and complaints to Letters Page, *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP.

Canoe in trouble

REFERENCE *Canoe, Dragon User* July 1986:

May I appeal for help through your columns with the above program, which was quite a lengthy one in machine code?

Unfortunately, the listing was incomplete: at address &H4A9A there is a 'JMP &H4724' instruction but there is no routine at &H4724! In view of the fact that no corrections appeared in later issues of DU, I can only assume that I was the only reader daft enough to type in such a mass of hex code, even if it did take nearly two years to get round to it!

Recent letters to Mr. Croucher requesting a listing of the missing code have failed to produce any response.

However, the article offered to supply tapes of the program. If any readers purchased such tapes and would send me a memory dump running from &H4724 to &H49FF I would be extremely grateful. Postage would naturally be refunded.

In closing, may I make the obervation that I find it distressing that a contributor should apparently take such a cynical attitude when asked to correct an error. Dragon users would be justifiably annoyed at purchasing a faulty program from a software house.

D. Swift
41 Headington Road
Maidenhead
Berks SL6 5LA

YES indeed and no indeed: you were certainly not the only person to type it in, as I remember a protracted correspondence with several people somewhat after Canoe **appeared. I seem to recall that the correspondence had a happy ending, and that Mr. Croucher played no part in it. Apart from that, my memory is shrouded in mist and I cannot for the life of me explain why this case was rather odd, or why no update appeared in DU.**

There may have been something in the nature of his having not actually written a routine at that location, and the program running satisfactorily with a minor bit of tweaking which somebody else worked out or saw at once.

There is nothing in the file. It is a mystery. Can anybody out there help Mr. Swift?

DU expects authors to sort out any bugs which crop up after publication, and in the rare case where we have had a seriously bugged program or very bewildered readers as a result of something we have published, the authors have waded in and sorted out. Unfortunately, in the case of a program which was published two years ago, we have added difficulties: there may be a perfectly innocent explanation for Mr. Croucher's silence; he may be of an age to be away at college or canoeing round the world; he too may have lost and forgotten his own program; or he may be so haunted by the terrible consequences of the missing routine that he has become a hopeless recluse.

On the other hand, if he or one of his mates is reading this, could you drop me a line to let me know? We aren't going to send the Dragon boot boys round to take the cheque back — honestly.

Every month we will be shelling out a game or two, courtesy of our supplies, to the reader/s who send the most interesting or entertaining letters. So send us your hints and your opinions, send us your hi-scores and suggestions. Send us your best Dragon stories. What d'you think we are, mind readers?!

EXTRA PUFF

DOS complexities raised

I would like to comment on some of the issues raised by Ken Smith in the May *Dragon User*.

Firstly, as SuperDOS E6 appears to be DragonDOS with about half the faults corrected, having a copy of the directory on track 16 is by no means 'safer'. Whenever a sector on track 20 is written, it is copied to track 16 when the motor times out. This means that directory corruption is also copied, usually before corrective action can be taken. I would contend that user-initiated backup (and recovery?) would be preferable, otherwise no second copy of the directory is required.

Next, SuperDOS E6 does not operate as described when using CLOSE, as the DragonDOS fault is still present. The cure for this, and a method of single file CLOSE operation, was given my DragonDOS corrections (DU May '85). Also, the idiosyncrasies mentioned at the end of the article are either introduced faults or some of the DragonDOS faults left uncorrected. I note there is no mention of data loss with FREAD, file corruption by DIR, undetected errors, erroneous errors etc. etc.

Finally, with a little effort and ingenuity, it is possible to produce a DragonDOS-compatible DOS (I currently run such a DOS) without the errors and with the features Ken Smith would like (SCOPY equivalent, selectable single/dual directory, single file close, etc.) and stay well within 8K. The remaining space can then be used for additions (cassette/disc program compatibility, named and numbered discs, etc.)

Philip G Scott, 4 Badgerwood Drive, Frimley
Camberley, Surrey, GU16 5UF

Tandy response

ON your letters page you said you would like to know how Dragon and Tandy owners would feel about you extending your repertoire into Tandy country. I can assure you that most owners would be very pleased. I think that if Dragon and Tandy owners get together a bit more, both machines will be in use a lot longer than they would trying to carry on individually. It would also help if software houses catered for both machines a bit more.

L. Gutteridge
51 Kyrle Road
London SW11 6BB

RAM on rampage

FOR some while now my Dragon 32 has had an annoying fault. The problem was lack of usable memory. When I bought my *Telewriter*, I could not load the Demo — there was insufficient space. I phoned Gordon Twist, who suggested that I PRINT MEM. This I did and up came 8487 (bytes). A loss of ram had occurred. I carried on with the fault for a long time.

Someone at work suggested that it could be a ram chip down, so I got my mate, who is a prototype wireman, to remove the sixteen 4116s (my board is an issue 5) and replace them with sockets. He took several lunch hours over the job, but in the end I had all my chips in their original positions. He then buzzed all the tracks and fitted links where the tracks were dodgy.

I took the board home, plugged it in, and turned it on. Sure enough, it worked, but I still had only 8487 of memory. I now took a new 4116 and fitted it in each position in turn, ignoring the first four, as the Basic goes up there, and the machine just crashes if it can't. At about the third from the end, I plugged the new one in and PRINTED MEM and lo and behold up came 24,871! I

checked in my book, and that is indeed the correct figure for a 32K Dragon. (The Basic takes up the remaining 8K).

I have enjoyed *Dragon User* for some years now, obtaining many useful tips from it and I am very pleased to be able to pass this one on to others who may be in the same boat as I was. I was given — in the course of my problems — an issue 2 board. It has a sam chip gone, but if anyone can make use of it I would be pleased to pass it on, for the cost of the postage. Finally has any one out there in Dragon land got the book *Inside the Dragon*, which may be lying around surplus, I would like one in exchange for Beer Vouchers.

Ian Roberts, G6HTZ
80 Mendip Vale
Coleford
Somerset
BA3 5PR

Disc hangups

I have been a regular reader of *Dragon User* for about four years now and have seen the quality of programs and articles improve greatly. If I have a problem I need only look in past issues and usually find the answer so I would like to thank DU, and of course all the companies that have continued to support the Dragon. I have found all the articles on DragonDOS very helpful, particularly Pam D'Arcy's 'Disc Rescue' which prevented me from pulling my hair out.

After having non-stop trouble with tape recorders, I decided to purchase a second hand disc drive. With the drive came a number of discs with some very good software, but one disc also contained files which I didn't want so I set about deleting them without including the file type. The result was a corrupted disc.

I then tried saving some of my own programs and got nothing but CC,RF,WP and IV errors. Come back IO error, I forgive you! Before I had pulled out too much hair I found the *Disc Rescue* program and was able to recover all but one program.

I think I have found some errors in the DOS as well as those published in your magazine, which other Dragon DOS users should know about. If DSKINIT is used in command mode the computer sometimes hangs up. If a file is KILLed and a new file of the same name is saved, I sometimes get a WP error. One occasion the computer hung up and corrupted the disc. On some discs the first file in the directory is corrupted when loaded.

Finally, one of my new discs developed a CC error on one file and an RF error on another where a file which was corrupted loaded alright. All this happened in the space of five minutes when I left the room and returned.

Desmond King
68 Farnham Street
Ormeau Road
Belfast
BT7 2FN

RAE course

HENDON College runs a very successful course for the Radio Amateurs Examination (City and Guilds 765).

Your readers may be interested in the course, which is held on Tuesdays from 7.30 to 9.30pm at Hendon College, Corner Mead, Grahame Park, Colindale, London NW9 5RA. The tutor is Tony Essex (*and not Grahame Park, as I was about to type — Ed.*) and the phone number for enquiries is 01 200 8300.

We also offer a range of full time and part time courses in electronics, including C&G 224 Electronic Servicing evening classes.

C. Holford
Senior Technology Llecturer
Hendon College
London Borough of Barnet

Basic09 print alarm

THANKS for printing my article *Basic09*. Due to the rather faint reproduction, plu probably the vagaries of my printer, I think it would be helpful if you print the following.

Basic09 (June 1988) contained a simple program as an example of what can be done. If anyone had a got at using it, they should be warned that the underscore which connects 'printer_path' did not reproduce clearly. In OS9, variable names can be pretty long, but no spaces can be left in them.

Perhaps this short note will prevent a few suicides.

Also in the final column of the article, the instruction 'Tandy/term' should look like 'Tandy/term' — the space is essential.

Best wishes to you and DU under the new management.

David Rothery
1 Heath Road
Glossop
Derbyshire
SK13 9AY

THANKS, David. Your listing was quite alright, apart from being dot-matrix, and although we always ask the printers to make the listings as contrasted as possible in that case, still very fine details, like the tails on commas, tend to bleach out.

In fact, the underscores do show in 90% of the lines, but it is possible that they could be overlooked and your warning is timely.

I hope that the underscore in your letter above is reproduced correctly — but owing to the vagaries of typesetting, underscores frequently get lost, an annoyance to people who write about languages like OS-9 or QL SuperBasic. It did get lost; we had to put it in specially.

Dragons A1 service

A few years ago when it became something of an inconvenience to share my BBC with my son, we decided to purchase a computer for him. At that point there was a particularly good sale offer at Dixons on the Dragon and, as an excellent job had been done to hide the oncoming future of Dragon Data, we were obviously not made aware of the problems in store, and went ahead with the purchase.

As it turned out, the problems were not so disastrous after all, but thanks almost solely to *Dragon User*. We subscribed to it immediately, which was to become our lifesaver and, had it not been for *Dragon User*, we would have changed to another computer ages ago. The main problem, of course, was the lack of suppliers for both software and hardware. It was a long time before we realised that by far the easiest way to get software and hardware was by mail order from people advertising in DU. After purchasing the initial games, etc. we decided to add a word processor and this was followed some time later (with equal ease) by twin drives.

Since my son (like his elder brother — a chief programmer) was interested in computers as a career, about 18 months ago we decided to extend his facilities by having an upgrade from 32 to 64. For this we contacted Compusense. They did a first class job and, considering it was close to Christmas, in very good time.

My son will be 18 in April and we agreed to purchase a monitor for his birthday, as he had been using his television until then. I thought at first that it might be more convenient to buy one locally and avoid the necessity of delivery costs but, as to be expected, when I approached the computer shops in Leicester, they did not want to know about the Dragon, with remarks like "Can't get you anything for the Dragon', "There isn't one for the Dragon' and all the rest. I should have known better by now so I went back to the faithful Compusense with dramatic results.

I phoned them on Monday March 21st to check their prices and availability and this gave me my first surprise. They said they would have to order one as they did not actually stock them and the price they quoted was exactly the same as twelve months previously.

I put a cheque in the post on Tuesday March 22nd and the monitor was delivered to my door at lunch time Thursday March 24th. The monitor came direct from the manufacturers and the cable came from Compusense. The cable arrived on Friday March 25th. This has to be the most efficient service I have ever experienced and I felt I simply had to write to tell you. If the pathetic British postal service ever goes public I pray it is someone like Compusense that takes over.

F. Brocklehurst
1 Park Avenue
Markfield
Leicester
LE8 0WA

News desk

If you have any new products for the Dragon — software or hardware — send us your press release

Diamonds and ghosts

DRAGONFIRE Services have launched a bouquet of new games in the £2 to £3 price bracket.

The new titles are *Diamond Manor* (£3.00), in which the challenge is to steal the Black Diamond from it's owner's house, *Sunken Ghost* (£3.00), in which you must borrow diving equipment and a boat and set out to find gold, silver and platinum in a haunted wreck, *Bust Out* (£3.00), an 100% machine code version of the classic *Break Out*, with variable bat and ball speeds and sizes, colour, music and a two player option, *Dragstone* (£2.00), an educational game for over-tens learning maths, history and biology, with an optional free copy of *Kids Pack* for the under-tens on the back if requested, and *Electra-Sketch* (£3.00), a hi-res drawing aid for users of the Tandy *Electronic Book* on the Dragon 32 or 64.

For details, post and packing costs and up-to-date price list, send an SAE to Dragonfire Services, 13 Parry Jones Close, Blaina, Gwent NP3 3NH.

Two pulses

PULSER Software's major releases of recent months, *Rally*, an 170K-plus game on DragonDOS disc only, and Jonathan Cartwright's *Utopia*, an air-attack/shooting game with detailed black and white graphics on cassette, are now both available, priced £7.99 and £5.49 respectively.

Enquiries to Pulser Software, 36 Foxhill, HIgh Crompton, Shaw, Oldham OL2 7NQ.

Orange appeal

ORANGE Software now have the licence to distribute DOS Plus 0.4 by Phil Scott (*not to be confused with Phil Stott — Ed.*) for DragonDOS, and its variant DOS Plus Delta, which allows the user to choose between DragonDOS and DeltaDOS either via a small hardware switch mod available from Phil Scott himself, or by loading.

Orange are also releasing the fifth adventure from Simon Hargrave, *Starke Island* priced at £5.00. It is likely, says Orange's Graham Smith, that they will run a special offer on the whole set of Hargrave adventures in due course.

Another new adventure from Orange is *War Hammer of Gillibran*, a long, alternative-choice style adventure on two discs. A price has yet to be established. "This is very long, but fairly straightforward", says Graham, "so it may appeal to arcade buffs as well as adventure players." He draws a parallel with the popular adventure books where the outcome is decided by the player's choice of actions, rather than by the solivng of riddles.

Tandy conversions of *Bean Stalker* and *Lucifer's Kingdom* are also available.

Information sheets from Orange Software, The Garth, Star Road, Nant-y-Derry, Abergavenny, Gwent NP7 9DP. Tel: (0873) 0435 55.

If it fits

Why have you published so much on Orange Software recently? People ask. Because they are a new company with a new, expanding list that Dragon User has not previously covered. All the news that's fit to print, remember. And they write and tell us what they are up to, without which, we wouldn't know, would we? It only takes ten minutes and a stamp to let the Newsdesk know what your new products and special offers are, so don't delay, write today, and don't forget to let us have a copy of anything you want reviewed.

Reviewers rule OK

YOUR editor has had one or two queries recently about reviewing for *Dragon User*, so this is a good moment to run through the guidelines again.

Whether or not present day Dragon users remember the days (not very long ago) when 90% of Dragon games were reviewed by Jason Orbaum, some folk remain under the illusion that only a selected elite is allowed to review for DU.

This is not true. It could be you.

All you need to do, to become a *Dragon User* reviewer is write a review which *Dragon User* decides, in its wisdom, to print, with your byline at its foot.

How do you go about this?

Observe carefully the style in which *Dragon User* reviews are written: as much pertinent detail as possible gleaned from a reasonably prolonged playing of the game. Good and bad features pointed out. An assessment of playability, value for money and appearance. Always include details of the supplier and the price at the top. Allocate between one and five Dragons, and explain why if you wish. Handwritten copy is acceptable for reviews, but please bear in mind that somebody has to copytype it.

No fixed rule

We don't set any limit on length; expect your review to be cut if necessary. Very long reviews are more likely to be trimmed than short ones, but there is no fixed rule. However, a dozen lines quoting liberally from the cassette inlay and concluding 'This game has great graphics and great sound and I think it is great.' is unlikely to make an appearance — we want to know what you actually experience when you play the game.

What should you NOT do to become a *Dragon User* reviewer?

Well, you shouldn't write to the Editor and ask to be sent software to review. Send your own reviews. If they are consistently good, and we print them, then one day, when you least expect it, a small Jiffy bag will appear on your doorstep enclosing the legend "Please will you review this, and as quickly as possible? If you don't want to, send it back, also as quickly as possible."

Surprise, surprise, we don't often get them sent back, although I do remember asking a well known software author to review a rather specialised program, only to discover that it was a direct competitor to one of their own.

Embarrassment, embarrassment.

If you don't review something that is sent to you, and you don't send it back, you don't get asked again.

Nobody is authorised by this magazine to write to a publisher saying that they are a *Dragon User* reviewer and asking for complimentary software. It is, of course, a free world, but complimentary review copies are sent out by the editor in person, and only reviews commissioned by the editor have a guarantee of publication.

On the other hand, everybody knows that *Dragon User* often publishes more than one review of a new program, and it is up to suppliers to decide to whom they wish to award review copies on their own account.

Reviewing for *Dragon User* is not, contrary to occasional opinion, a way to build a large free software collection. There is a limited amount to go round; it is shared out according to the magazine's requirements, and the editor's decision is final. It may come as a surprise to many readers that over half the reviews published, on average, including those from 'regular' reviewers, derive from programs bought by the reviewers for themselves.

Indeed, it is generally true that without the enthusiasm to collect and refine one's critical judgement simply for the love of it, a person will never acquire the knowledge and conviction which leads to regular publication in a particular field.

New software for review should be sent to *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP.

One to three and climbing

Program: *Destiny* trio (disc version)
Supplier: Pulser Software, 36 Foxhill, High Crompton, Shaw, Oldham OL2 7NQ
Price: £8.99 the set

WELL, you adventure players, here is a three-in-one disc for you to savour. The first item is *Starship Destiny*, a maze-type adventure. The location of the 'maze' is an alien spaceship which has entered the galaxy and seems to be here to destroy the earth. The object of the game is to disarm the alien missiles and to get out alive. The game opens with the player being teleported into a carg bay in the alien ship. As the adventure is played in real time, after you have disarmed the missile you must teleport home, and if you linger too long in any one location a patrol will arrest you and dump you in confinement. You can of course escape if you have the keys, otherwise you are stuck. If this happens and you cannot think of a solution you may as well abort the game and start again (*Have you tried standing on a dwarf? — Ed.*).

Having described the game I found it a little frustrating because, despite being told that the game is in rea time, the screen images took too long for my personal taste to disappear and re-appear, and anxious to get on or see through my anticipated solution the wait seemed intolerable, although in actual fact it is only seconds.

The overall impression of this adventure is one of a 'good game but lacking polish'. It would be improved if it was auto-run and quicker, as the pressing of the enter button continually increases the sense of waiting for the game. The screen display would have been enhanced if changes in colour had been introduced. The black, yellow and white areas give every appearance of being dreary, although the game is not.

The second game, *Dungeon Destiny*, devised by Tim Riley and written by Jonathan Cartwright, is a distinct improvement on *Starship Destiny*. It is still an adventure maze, but the graphics are far superior and the adventure itself also compares very favourably, although the instructions had to be input in the lower case, which for me detracted from the display. Any attempt to press the shift key resulted in a negative response from the computer. "You can't .... that here" seemed a frequent response, and the continuing display of "What now, oh brilliant one?" grated after several repetitions, like most forms of 'wit' which are only funny if they come as a surprise.

Notwithstanding these minor criticisms, I nevertheless throughly enjoyed this adventure, and if I had only £3.99 to spend I would choose this one rather than *Starship*.

The third game is *Wildwest Destiny* which for me was the best. However, when I first tried the disc I thought that it had corrupted, because I made the mistake of loading without sound. The screen display told me not to remove the disc, and nothing seemed to happen. I tried again with sound, and the computer played a tune for almost 90 seconds before it actually loaded — however, one criticism of the earlier games was overcome: it did auto run. Once commencing the game I found that I had been killed in about six moves flat without warning, but at least it gave me another game immediately. Into the second game after trying the usual commands like enter, drink, etc., for reasons of speed I tried Help, which told me, among other things, to be prepared for anything and one certainly must. The second game lasted longer — just! If you enjoy your frustrations, this game is definitely for you.

Overall I enjoyed this disc and in terms of hours at the keyboard it is very good value. Each of the three games is better than the one before, and the development of the programs can be clearly seen. For those who specialise in adventures, this may be a bit 'old hat', but for all-rounders it is certainly worth a try.

Roland Hewson



A word could tell a thousand pictures

Program: *Picture Maker*
Supplier: John Penn Software
Price: £5

I was asked to take another look at *Picture Maker* after Gordon Lee complained that the instructions were difficult to find and use. Well, unfortunately, for David Makin, the author, I must concur with Gordon Lee on many points. The instructions, which are the only problem, are on two sheets of A4 paper in Lilliput type, so in order to make things a little easier for myself I ran the instructions from the program (where they are available) through my printer and received the same information, minus the screen dump routine.

The whole object of a set of instructions is to allow the absolute beginner to understand them, carry out the necessary commands and produce the object for the activity. (*Pardon, Roland??*). The problem inherent in *Picture Maker* is that the program is extremely sophisticated, and the instructions do not give a working example with which to practice. Utilities are there to provide the ordinary computer user with a facility that he can make use of fairly readily, and *Picture Maker* does not yet meet that criterion. The two pages of instructions, although detailed, are not comprehensive nor easily read. There have been other utility programs in the past for creating graphic screen displays; one that sprigns to mind is *Graphic Writer* by B R Smith, which can be run in Pmode 0, 2 or 4, and is less sophisticated than *Picture Maker*, but comes complete with a fourteen-page instruction booklet written in single, easy to follow routines that are logical and sequential. *Pixel Editor*, which can also be used to create and edit graphics shapes in modes 3 and 4, contains the equivalent of five pages of fine A4 print for its instructions, complete with an example.

It would appear that David has put a very great deal into *Picture Maker*, but it cannot be accessed and used to its full because of his assumption that the user will find it easy to follow, which is not the case.

However, I was impressed with the functions of the program itself. The ability to move areas about, to delete and replace, and indeed even to stretch stored areas horizontally or vertically is all there. Alterations are automatically re-numbered, and you can store up to eight different areas at a time depending on their size, which needs to be remembered when attempting to stretch an area, because obviously if the resulting area is then too large or all eight areas have been used, there will be insufficient room and the program will ignore the function.

A most impressive command is the M for magnify, which magnifies the area under the cursor position on the normal viewing screen to a magnified version on a different screen. The size of the area magnified (not the magnification itself) can then be enlarged by a simple key depression, and colour change too set at the touch of the keyboard. The magnification screen is then updated even when not in use, which allows the speed of movement to be controlled. To couple all these functions to a true 'picture maker' which will draw lines, colour in a foreground colour from the cursor position or a border in the background

Continued on page 11

DRAGONSWORD!

Paul Grade takes a monthly stab at setting the world to rights

ON the 20th of last month I received a letter from Her in charge informing me that the next deadline was the 20th of the month, and as I happened to be in the middle of trying to answer a couple of dozen letters, repair an overworked photocopier, and get the NDUG newsletter ready at the time, it didn't really seem like a practical proposition to me. Hence the delay. (*I was only teasing, Paul — sorry about the coronary*).

Of course, I haven't a clue what I'm supposed to be writing about. Reviewers can review, technical writers be technical, and Editors can write Editorial Comments, but what does that leave for me? Usually I only write for Update, which makes the problem easier because it's my own publication and I can say what I like without the risk of anyone editing out all my best insults.

It would be nice to be able to fill pages with details of all the new firms rushing to support the Dragon, with reports on all the newest hardware and software releases, but they would have to be very small pages and I'd have to use very large type.

Yes, I know Dragon Data did a lousy job of marketing the machine, and a lot of software companies imagined that people would pay silly prices for rubbish, but that is only a contributory cause of The Decline and Fall of the Dragon. The real reason was, and still is, that far too many computer owners are dedicated followers of fashion. You don't believe me? How many former Dragon owners do you know who have switched to other machines? A couple of years ago they were ditching the Dragon in favour of the BBC, a little later it was the Master or the Amiga, and currently it's Amstrad, ST, or one of the cheaper PC clones, right? Now for the trick part of the question: how many of those who changed machines had actually taken the trouble to learn to use the one they already had?! Of the ones that I know, less than 1% had a genuine reason to change. They mostly hadn't even managed to find a use for the memory available on the Dragon, but they were sure that they 'needed' at least 512K of ram, and I'm sure you've all heard the old excuse about needing a "better machine for business', mainly from people whose accounting begins and ends with reading their Access statements.

If they needed more memory there was no reason why they couldn't use the space available on disc; after all, that's exactly what most of the 'business' systems do, there's no need to have several miles of program resident in ram all the time. I admit that there are exceptions, and that some people have taken the Dragon to its limits and found it still doesn't give them what they need, and in those cases a change is justified, but they are a small minority.

So what does all this have to do with you? Quite a lot, because you subscribe to *Dragon User*, and the reason that Sunshine eventually decided that DU had to go was that subscriptions had dropped to a figure that was too low for them pay their contral London company expenses for the magazine. Now, Bob Harris writes and markets good Dragon software, and in taking over DU he is acting in self defense as much as anything else. If the Dragon market gets any smaller it will just not be practical for Bob or anyone else to produce software for it.

Before you rush down to Dixons or where ever, with your plastic money melting in your sticky little paw, try thinking. This is quite a simple process, and although it can cause some discomfort there is no actual danger, not even a Government Health Warning, so it isn't likely to do any lasting harm. Try thinking about why you have a computer, what you need it for, and what advantages there are, if any, in spending a medium-sized fortune on changing to a different machine. There isn't much that a Dragon can't do that other machines can, and do you really need ST graphics when you haven't even managed to draw a picture with the machine you have? Do you really need *Wordstar* or *Lotus 1-2-3*? Do you really want an expensive, virus ridden lump of plastic that's a nightmare to program, runs commercial software that needs a second mortgage to buy, and operates on MS-DOS and CP/M, both antique when the Dragon as first invented?

More to the point, are you really so anxious to admit that you can't even learn to program one of the easiest machines ever made? That keeping your overdraft up with the Joneses' is more important than sticking with a machine that can do what you need?

OK, don't get the underwear twisted, I know I'm breaking the rules by criticising the punters, but look at it from my point of view for a moment. Four years ago I started the NDUG, and we've currently got around a thousand members. *Dragon User* has been around even longer, and on a good day has about three times that number of subscribers, and the 'local' Dragon user groups can just about make up another 200 between them. Some of our suppliers have mailing lists stretching back years, but how many people on those lists are actually buying software?

You don't need a computer (do you?) to work out that this doesn't really add up to more than five thousand Dragons actually up and kicking. Total UK sales of Dragon computers were very close to six figures, and even allowing for natural wastage (Dragons which were burnt out, blown up, sat on, or whatever) that still leaves a lot of machines unaccounted for, doesn't it? Some are probably owned by 'rugged individualist' types who don't need groups or magazines (*More than is generally imagined, according to my knowledge of rugged individualists — Ed.*) but what happened to the rest? Are they all being used as door-stops or to prop up the bed? My guess is that most of them were consigned to attics and cupboards because the owners either got tired of apologising to Commodore owners for owning them, or felt that they simply had to have the latest in over-advertised status symbols.

If this is correct then there is still a lot that you could do to give the Dragon a boost. Most of the status machines sell on one of two points, ridiculously large memory and/or good (if expensive) software. Sure there's the usual 'hype', but that's not really very important. Now there's not a lot you can do about boosting the Dragon's RAM space. It can be hiked up to 512K without too much trouble, but that isn't going to persuade the types who relegated their Dragon to attic when they bought the ST to get it down again. However, if there was more software available (and advertised!) which could put the Dragon ahead of the current machines I think you'd find a lot of Dragons coming back out of the woodwork. Programmers aren't some peculiar (well, some of them are) lifeform specially bred by the Atari Corporation, most of them are people like you; they could be you.

You don't need to be any particular age or sex, or even some kind of genius, in fact it's probably better if you aren't a whizz-kid, computers are very simple minded machines, and they don't always take kindly to 'clever-clever' programmers.

I get tired of answering the same old letters which moan about lack of software, lack of support, etc. Anyone owning a machine is capable of learning to write good software on it, and those with Dragons have a big advantage over most others in that they have one of the best 'programmers' machines' ever produced, so why are you reading this instead of writing programs?! Of course, it isn't only the software side that things could be done to make the poor old beast look more attractive to the fashion followers. One of the current crazes is for frame grabber/digitisers. They aren't difficult to make, and even old heaps like the old Commodore 64 had them available, so why doesn't someone design one which is not dedicated to a particular machine, one which could be used with the Dragon and other machines with a change of software? I know that a lot of Dragon owners are good electronics engineers, some of them even owning their own companies, so why don't they help themselves and the Dragon by giving it a try? The Dragon's strong point is, that it does not require 'dedicated' peripherals, it can run any make of printer or disc drive or tape recorder, it isn't a fussy machine at all, so what's wrong with a 'universal' design which only needs different driver software to adapt it to several machines? That would make the Dragon a serious rival to most of the current machines, with the added advantage of being a lot cheaper! OK, end of lecture, but why not give it a try?

Paul

64K in two 32K pages

P D Smith consults Motorola about the real SAM

I have written this article to complete the information on the SN74LS783 (MC6883) sam chip given by Matthew Lodge in the August '87 issue of *Dragon User*. In his article Mr Lodge says that he doesn't know what the sam register at $FFD4 and $FFD5 does. I have just received the 74LS783 data sheet from Motorola and this clarifies the use of this register.

The register (referred to as allows you to use the 64K ram in a 64K6809 computer (such as Dragon 64) as two separate 32K pages addressed from $0000 to $7FFF. In fact what happens is that, in map type 0 (ROM at $8000-$FEFF) the A15 bit of the ram addresses is replaced by the 'P' bit from the sam. This means that if 'P' is reset (zero) then any access to the ram will access the obvious byte. For instance if you access byte 10000, you will use byte 10000 of the ram. However if 'P' is set (equal to one) then any access to the ram will access the byte in the second 32K page, ie accessing byte 10000 will in fact use byte 42768 (10000+32768). This is really quite difficult to explain and is best understood by some experimentation.

'P' is altered by accessing sam register P1, as can be seen from **table one**. So to set 'P' you must POKE address $FFD5 with any value. To reset 'P' you must POKE address $FFD4 with any value.

Check the mode

There are several things to note about this: 1) this paged mode can only work from the map type 0, that is, the 32K mode of a Dragon 64 2) the paged mode will only work if you have 64K in your computer. If you only have 32K the 'P' bit will have no effect as the A15 bit of the address is not used 3) if you set the 'P' bit without preparation the computer will immediately crash (as some of you will have found out by now). This is because the computer will still be generating interrupts which use ram based vectors, and the Basic interpreter uses vectors in ram. Because these vectors have suddenly disappeared the computer will undoubtably crash.

The first program is the answer to the problem of the computer crashing. This copies the whole of ram into the second ram page, so that when the pages are swapped, Basic will continue as if nothing has happened. Note that the contents of $7F00 to $7FFF cannot be copied into $FF00 to $FFFF as these latter addresses are reallocated by the sam to I/O, the sam and the interrupt vectors. However locations $7F00 to $7FFF can be used one you are in the new page of memory.

Before you run **listing one** you must have set the top of memory to less than $7F00, otherwise the system stack will vanish and the computer will crash. To test the paging system you must now set the 'P' bit by 'POKE&HFFD5,0'.

```
2700                *PROGRAM 1
2700
2700                *COPY RAM ($0000-$7EFF) TO
2700                *RAM ($8000-$FEFF)
2700
2700                *NOTE - CANNOT COPY TO RAM
2700                *$FF00 TO $FFFF AS THIS IS
2700                *USED FOR I/O, SAM AND VECTORS
2700
2700 3401           @COPRM PSHS CC
2702 1A50                  ORCC #$50   ;DISABLE IRQS
2704 7FFFDF                CLR $FFDF   ;SELECT RAM
2707 8E0000                LDX #0
270A 108E8000              LDY #$8000
270E
270E EC81           COPYLP LDD ,X++
2710 EDA1                  STD ,Y++
2712 8C7F00                CMPX #$7F00
2715 26F7                  BNE COPYLP
2717
2717 7FFFDE                CLR $FFDE   ;SELECT ROM
271A                *RESTORE IRQS AND RETURN
271A 3581                  PULS CC,PC
271C
```

```
10 ' BASIC PROGRAM 1
20 CLEAR 200,&H7EFF
30 FOR I=3072 TO 3072+27
40 READ A$
50 A=VAL("&H"+A$)
60 CS=CS+A
70 POKE I,A
80 NEXT
90 IF CS<>3293 THEN PRINT "ERROR IN DATA":
   SOUND 10,10:END
100 PRINT "DATA OK"
110 EXEC 3072
120 DATA 34,01,1A,50,7F,FF,DF,8E
130 DATA 00,00,10,8E,80,00,EC,81
140 DATA ED,A1,8C,7F,00,26,F7,7F
150 DATA FF,DE,35,81
```

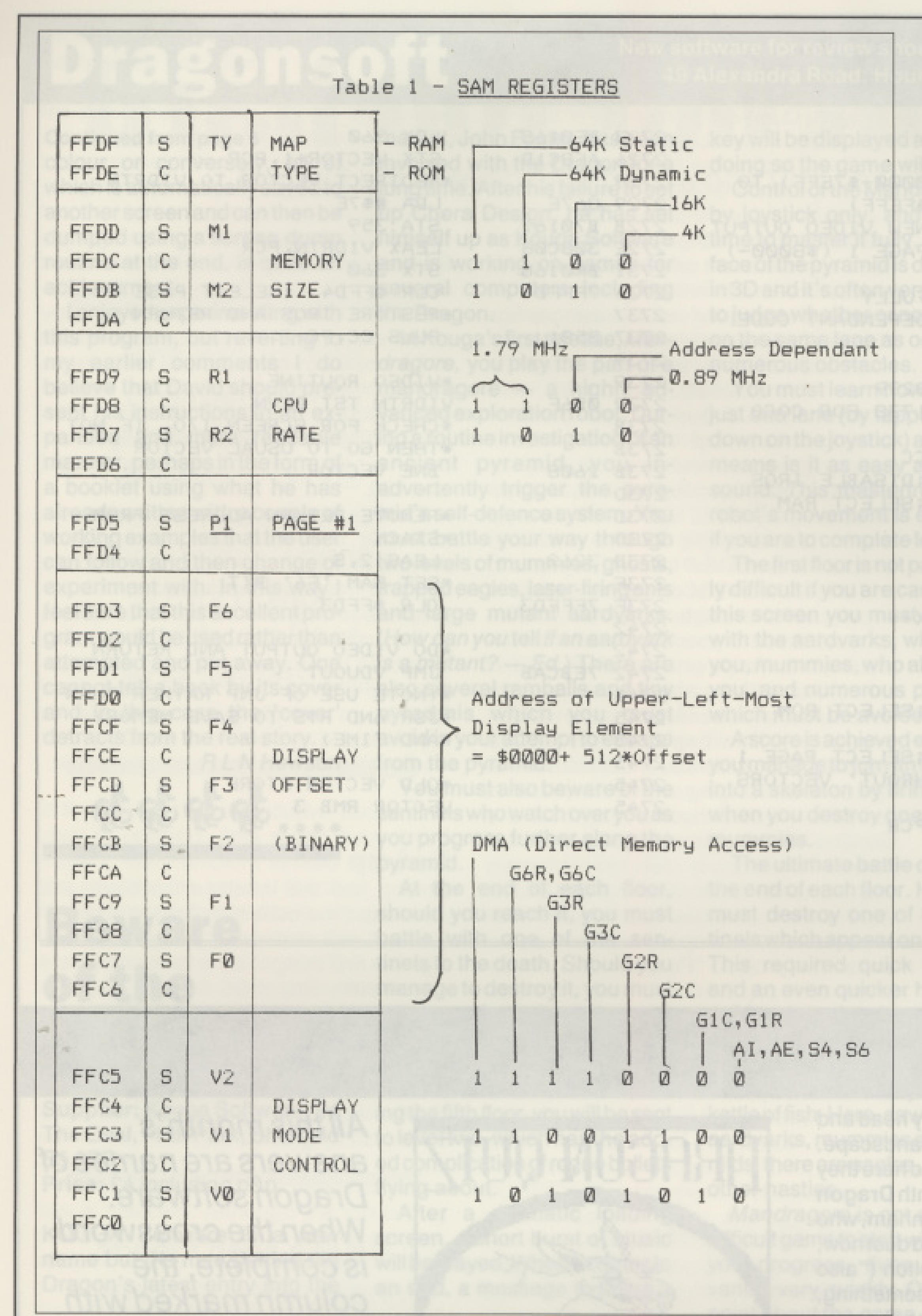
```
   BASIC PROGRAM 2
 10 POKE &HFFD3,0
 20 GOTO 20
```

The computer will now appear to have crashed, but if you try a sound command such as 'SOUND 1,1' you will hear a beep showing that the computer is still working.

The reason that nothing appears on the screen is that the sam is still displaying 1024-1535 as the text screen, while Basic is writing to 33792-34327 (but still thinks that it is writing to 1024-1535). In theory this problem can be solved by telling sam to display the correct addresses, by setting F6 (see **table one**). This will work but not for very long, as Basic will instantly reset it when the next character is to be printed.

To see that it can be done, try typing in Basic **listing two**; this will have to be done 'blind', but it is short enough to be done. When you run this, you will see all the text that you have entered in your panic when you thought the computer had crashed. Now if you press BREAK the computer will redisplay 1024-1535. (A useful note here: *Dream* does not reset the text screen when it is run (at least mine doesn't) so *Dream* can be easily used in this mode.)

If you now reset the 'P' bit by 'POKE &HFFD4' you will see that the screen is working again. Also note that program two



has vanished. If you set 'P' again, **Listing two** will be back.

Now the main problem is caused by the BASIC PRINT routine resetting the text screen. **Listing three** solves this by intercepting the PRINT command and missing out the 'RESET VIDEO' routine. The graphics screens will still be incorrectly displayed, but this can be corrected by setting 'F6' manually as Basic does not reset these.

An important point can be seen from **listing three**. If you have written a program to swap between the ram pages, then the program will have to exist in identical positions in both pages. This is so that the program does not vanish when the pages are changed. Also note that, since the memory is changed, you cannot pass variables on the stack (or in any other portion of memory) between the two pages. The only way to pass variables between the pages is to loan the variable into a 6809 register and then swap pages.

Listing 3a

```
10 ' BASIC PROGRAM 3
20 CLEAR 200,&H7DFF
30 FOR I=&H7E00 TO &H7E00+68
40 READ A$
50 A=VAL("&H"+A$)
60 CS=CS+A
70 POKE I,A
80 NEXT
90 IF CS<>8183 THEN PRINT "ERROR IN DATA":SOUND 10,10:END
100 PRINT "DATA OK"
110 EXEC &H7E00
120 DATA 34,01,1A,50,7F,FF,DF,8E
130 DATA 00,00,10,8E,80,00,EC,81
140 DATA ED,A1,8C,7F,00,26,F7,7F
150 DATA FF,DE,7F,FF,D5,B6,01,67
160 DATA A7,8C,22,BE,01,68,AF,8C
170 DATA 1D,86,7E,B7,01,67,30,8C
180 DATA 08,BF,01,68,7F,FF,D4,35
190 DATA 81,0D,6F,26,08,32,62,7F
200 DATA FF,D3,7E,BC,AB
```

Even though I have found out about this paging function of the sam, I cannot think of any uses for it, but maybe someone using a Dragon was thinking "I wish I could use the 64K as two pages of 32K instead of having to go to map type 1 (64K mode)'. Well, now you can.

The real advantage of this paging mode will be apparent if you can program eproms and so have a 32K program in eprom and have 64K of ram as well. In this case you have 96K of memory from a computer which should only allow 64K. You could also use a static ram in place of an eprom (either in place of the second Basic eprom from a Dragon 64, or in a cartridge).

Beware speed pokes

To finish I will give a comment about the 'speed up' pokes for the Dragon. Matthew Lodge, in his article about the sam, says that there are four speed modes given by the sam. In fact there are only three. The two he refers to as 'superfast' and turbocharged' are identical, with a clock speed of the crystal frequency divided by 8, ie 1.79 MHz. You should only use these 'speed pokes' if the 6809 microprocessor in your Dragon is capable of a clock speed of 1.79 MHz.

The actual microprocessor in the Dragon is the 6809E, which has a maximum speed of 1 MHz. There are two enhanced versions available, the 68A09E and the 68B09E which allow clock speeds of 1.5 MHz and 2.0 MHz respectively. (The 'E' after 6809 means that the 6809 uses an external rather than an internal oscillator). This means that to run at speeds of 1.79 MHz you should really use a 68B09E. The 6821 pias should also be replaced by 68B21s otherwise you could damage them by using them at the higher speeds. The reason that some Dragons will work satisfactorily at the higher speeds is because all manufacturers tend to 'underrate' their products, so that a 6809 may work at 1.8 MHz, but is not under guarantee if used above its specifications.

Listing 3

```
2700          * PROGRAM 3                    2723 BE0168       LDX 360
2700          *                              2726 AF8C1D       STX VECTOR+1,PCR
2700          *COPY RAM ($0000-$7EFF) TO     2729              *REDIRECT VECTOR TO VIDRTN
2700          *RAM ($8000-$FEFF)             2729 867E         LDA #$7E
2700          *AND SET UP NEW VIDEO OUTPUT   272B B70167       STA 359
2700          *VECTORS IN PAGE 1 ($8000-)    272E 308C08       LEAX VIDRTN,PCR
2700          *                              2731 BF0168       STX 360
2700          *WRITTEN IN FULLY              2734 7FFFD4       CLR $FFD4 ;SELECT PAGE 0
2700          *POSITION INDEPENDANT CODE     2737              *RESTORE IRQS AND RETURN
2700                                         2737 3581         PULS CC,PC
2700 006F     DEVN EQU 111                   2739
2700 BCAB     VDUOUT EQU 48299               2739              *VIDEO ROUTINE
2700          *SHOULD BE 41738 FOR COCO      2739 0D6F         VIDRTN TST DEVN
2700                                         273B              *CHECK FOR SCREEN I/O, IF NOT
2700 3401     @VIDOT PSHS CC                 273B              *THEN GO TO USUAL VECTOR
2702 1A50       ORCC #$50  ;DISABLE IRQS     273B 2608          BNE VECTOR
2704 7FFFDF     CLR $FFDF  ;SELECT RAM       273D
2707                                         273D              *REMOVE RETURN ADDRESS FROM
2707            *COPY RAM                    273D              *STACK
2707 8E0000     LDX #0                       273D 3262         LEAS 2,S
270A 108E8000   LDY #$8000                   273F              *SET SAM 'F6' BIT
270E EC81       COPYLP LDD ,X++              273F 7FFFD3        CLR $FFD3
2710 EDA1       STD ,Y++                     2742
2712 8C7F00     CMPX #$7F00                  2742              *DO VIDEO OUTPUT AND RETURN
2715 26F7       BNE COPYLP                   2742 7EBCAB       JMP VDUOUT
2717 7FFFDE     CLR $FFDE  ;SELECT ROM       2745              *(NOTE USE OF JMP RATHER THAN
271A                                         2745              *JSR AND RTS TO SAVE MEMORY
271A 7FFFD5     CLR $FFD5  ;SELECT PAGE 1    2745              *AND TIME)
271D            *SAVE OLD 'CHROUT' VECTORS   2745
271D B60167     LDA 359                      2745              *OLD VECTOR STORE
2720 A78C22     STA VECTOR,PCR               2745              VECTOR RMB 3
```

Crossword

The ninth Dragon Crossword raises its hairy head and looks around in bewilderment at the new landscape. "My fans must be here soon", it thinks — and here they are — oh, no, this is the fan club of the seventh **Dragon User** crossword, Tony Arkwright of Chippenham, who would like a recent Microdeal game (like gold dust now, those are, my boy) and G. Brenner of Bolton ("also known as Leeds") who wants to win something, anything ... quick, pass one of those spare tea chests...

The phrase is THE DRAGON USER.

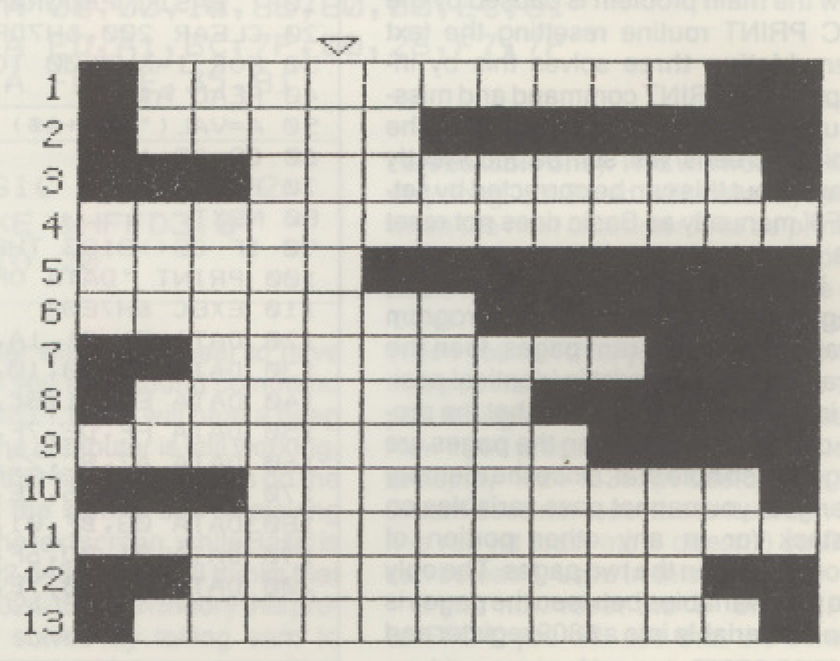
There will be a couple of free tapes from the Editor's Magic Bottomless Box for the first correct entries out of the hat each month. You can try telling us which tapes you'd like — you never know, we may have them.

And you don't have to cut up your *Dragon User* — entries on a photostat or a plain piece of paper will do, as long as we can read them.



All this month's answers are names of Dragon software. When the crossword is complete, the column marked with an arrow will spell out a phrase.

1 and 10. Initially the butcher goes on safari (8,2,3,6)
2. Sounds like this man avoided an excavation! (2,3)
3. Can inroad change into harsh rule? (9)
4. 'The Eagle', firm, proceed! (5,6,2)
5. Dashes to the game (5)
6. Regal agony for these invaders (7)
7. Film about a gloomy celebrity (4,4)
8. Crazy for 3.14159! (7)
9. Tops in football (9)
10. See 1.
11. Is destiny decided in this large house? (7,2,4)
12. Return and do this if you get lost (9)
13. Assault by the spirit! (5,6)



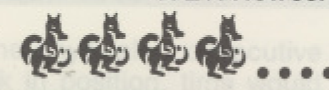
Dragonsoft

New software for review should be sent to *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP

Continued from page 6
colour or conversely, all of which is automatically saved to another screen and can then be dumped using a screen dump routine at the end, is quite an achievement.

I enjoyed experimenting with this program, but reverting to my earlier comments I do believe that David should present his instructions in an expanded and more readable manner, perhaps in the form of a booklet using what he has already written with a couple of working examples that the user can follow and then change of experiment with. In this way I feel sure that this excellent program would be used rather than attempted and put away. One cannot tell a book by its cover and in this case the 'cover' detracts from the real story.

R L N Hewson



Beware of the Aardvark

Program: *Mandragore*
Supplier: Kouga Software, 94 The Oval, Firth Park, Sheffield S5 6SP
Price: £4 including p&p

KOUGA Software is a new name but the man behind the Dragon's latest entry into the market, John Foster, has been involved with the Dragon for a long time. After his failure to set up Chera Design, he has set himself up as Kouga Software and is working on games for several computers including the Dragon.

In Kouga's first release, *Mandragore,* you play the part of a mandragore — a highly advanced exploration robot. During a routine investigation of an ancient pyramid, you inadvertently trigger the pyramid's self-defence system. You must battle your way through two levels of mummies, ghosts, trapped eagles, laser-firing ants and large mutant aardvarks. (*How can you tell if an aardvark is a mutant? — Ed.*) There are also several ramballs and tiny pyradmis which you must avoid in your attempt to escape from the pyramid.

You must also beware of the sentinels who watch over you as you progress further along the pyramid.

At the end of each floor, should you reach it, you must battle with one of the sentinels to the death. Should you manage to destroy it, you must fall down the hole to the next floor where your mission will continue. In all there are five floors on level one. On completing the fifth floor, you will be sent to level two, which has the added complication of rogue bullets flying about.

After a dramatic loading screen, a short burst of music will be played. When it comes to an end, a message to press a key will be displayed and upon doing so the game will start.

Control of the Mandragore is by joystick only, and it takes time to master it fully. The surface of the pyramid is displayed in 3D and it's often very difficult to judge whether or not you are on the same lane as one of the numerous obstacles.

You must learn how to move just one lane (by tapping up or down on the joystick) and by no means is it as easy as it may sound. This mastering of the robot's movement is essential if you are to complete level two.

The first floor is not particularly difficult if you are careful. On this screen you must contend with the aardvarks, who fire at you, mummies, who also fire at you, and numerous pyramids which must be avoided.

A score is achieved each time you manage to turn an aardvark into a skeleton by firing at it or when you destroy one of many mummies.

The ultimate battle comes at the end of each floor. Here you must destroy one of the sentinels which appear on the wall. This required quick thinking and an even quicker hand.

After dropping down the hole at the end of each floor, you continue to the next floor.

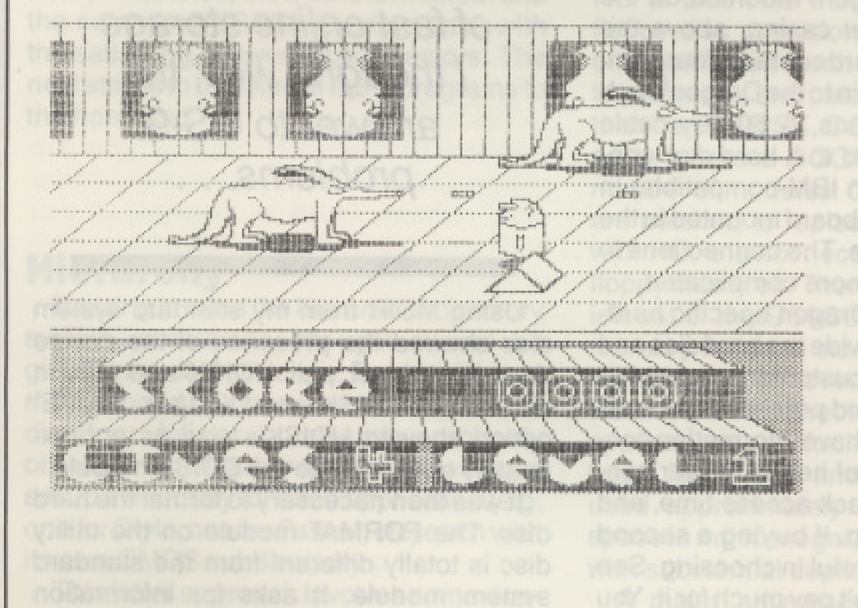
Level two is a totally different kettle of fish. Here, as well as the aardvarks, mummies and pyramids, there are eagles, ants and other nasties.

Mandragore is not a terribly difficult game to start with but as you progress, lives start to vanish very quickly. One good point about the game, though, is that when you are killed, you continue where you left off.

In all you have five lives and while this may seem a lot, it's suprising how quickly they disappear.

The graphics are in 3D (for the most part) and are breathtaking. Never before have I seen a game achieve graphics of the standard of *Mandragore* (*Phone the Guinness Book of Records, quick.*). They really have to be seen to be believed. As you progress further, they get better.

The screen is displayed in PMODE4:SCREEN 1,0 graphics. Perhaps PMODE4: SCREEN1,1 graphics would have achieved a better degree of sharpness. As it stands, though, the screen contents are still a marvel.



The music is well written and the sound accompanying the game is of the highest standard. I have nothing but praise for this masterpiece. It is also a very challenging game which should challenge hardened arcade fanatics to the limit. I can say with hand on heart that rarely will you find a game as good as this one for under £5.00.

Donald Morrison

Imperfect world is nearly a perfect game

Program: *Utopia*
Supplier: Pulser Software
Price: £5.99

UTOPIA is another game written by Starship Software, also repsonsible for *Rola Ball* and *Spy Against Spy*. But these pale into insignificance compared with *Utopia* their latest game and undoubtedly their best.

A Utopia is "an imaginary state with perfect political and social conditions or constitution". Until recently (in the game scenario) the man made colony of the same name was just that. The Galactic Federation built it up into an idea artificial planet, but now it has fallen into a state of disrepair and as a result is no longer the perfect place it used to be.

Unfortunately, however, the inhabitants of Utopia are unable to leave the planet because of its complex defence systems. Originally they were designed to stop intruders from entering the colony, but what the Federation failed to realise was that the people who lived there were as a result trapped, with no means of escape.

The task of escaping from Utopia has been given to you. As a member of thee rebels, you must guide three of the most advanced star fighters, stolen from the Federation, through Utopia's complex defence system to freedom.

The three star fighters in your possession are so advanced

Continued on page 19

Hard but not too hard

***Roger Merrick** links OS-9 and a hard disc drive to the Dragon.*

WHEN I previously wrote about my impressions of *Stylograph*, I shared some of my frustration on using OS-9 on the Dragon. My criticisms/comments should not be taken to imply dissatisfaction with OS-9, or the Dragon, but with the combination of the two. Having worked with it now for some years, but not claiming by any means to be an 'expert', I think. OS-9 is a wonderful operating system, but using it on the Dragon I have encoutered a number of problems that I find very irritating.

Floppy discs

I suppose I should be grateful that OS-9 no longer uses single density drives, and I should be 'thankful' that I do not use a CoCo with only 35 tracks per disc. I do recognise that back in the good old days of 1983, a single sided 40 track was the most cost effective solution to providing a floppy disc system for the Dragon. But I feel cheated, in a way, that nobody ever made it clear that two drives were really essential for OS-9. Of course, Dragon Data had some terminal misfortune after the release of OS-9, and it was fortunate that they got the OS-9 software released before the collapse. That was in fact the last benevolent act of a stricken computer company who really did appear to care about their customers (and this has perhaps been repaid by the fact that we are still here five years later). I have been told that Microware never received the licence fee from Dragon Data for the OS-9 software, so there were no 'legitimate' issues. So in issuing Version 2 to registered users, Microware were also being benevolent.

However, all that being so, I nevertheless find the standard double density, single sided floppy drive does not provide enough space to store a reasonable set of commands, an application and data on the same disc. Single drive operation is extremely time consuming, requiring modules to be loaded in memory, data directory to be changed, etc. Working discs require preparation, and constructing customised BOOT discs using OS9GEN on a single drive system is difficult.

One simple solution is to add a second drive. If this drive matches the original, then online storage is doubled. The original Dragon Data release of OS-9 only supported the single sided 40 track format. V2 from Eurohard includes disc device descriptors and drivers which support double sided and 80 track formats as well. A double sided 80 track disc quadruples online storage, offering .75Mbytes per disc. But floppy disc operation in OS-9 is still unsatisfactory — the type-ahead buffer stops working while the disc is being accessed. At first, I took this to be poor software writing in the device drivers, but now I come to see it as a problem with the hardware. Also, maintaining a reasonable amount of workspace means having as few modules permanently resident in memory as possible. This emphasizes an operational characteristic (for this read 'drawback') of OS-9 — the time spent accessing discs to load modules.

Having seen friends and colleagues with PCs, using hard disc systems that provided megabytes of fast online storage, and envied, I thought, well, the answer to the problems of OS-9 on the Dragon is to add a hard disc. The operating system is sophisticated enough to handle it. Then you could have all your commands and applications available at the touch of a 'CHX'.

Knowing nothing about hardware constructions, I searched for a supplier. I drooled over the 'off the peg' systems available for the CoCo in America, and shuddered at the cost of bringing one over (with no guarantee that it would work on the Dragon). Compusense at one time appeared to have one planned. However, their plans for 're-establishing' the D64 at £299 weren't the howling success that might have been hoped, and they abandoned the project. "People would not be prepared to pay the cost of such a system, sir", they told me. So, at the last 6809 show at the Horticultural Halls, I asked Martin Vernon if he knew anyone who could do a hard disc interface. Gordon Twist's name came up, and some time later I contacted him.

Hard disc system

The hard disc system consists of the following subsections: the dragon interface board, hard disc contoller board, power supply, and hard disc driving software.There is a small interface board that is fitted inside the Dragon, mounted on the shoulder of the lower casing, above the reset button. This board contains four components, and is wired in to the Dragon's pcb by several connections. A 50-way cable leads out of the board to a hard disc controller board (from an IBM-compatible) in the next section. The board mounted in the Dragon is very simple. The connections to the Dragon pcb are more complicated.

But this is all the Dragon-specific hardware. Gordon will provide the hard disc and power supply, or the customer can provide their own. If you intend purchasing a hard disc, ensure that you have information concerning the number of heads, power consumption, track to track access time, and damaged sector map. If buying a second hand drive, do be careful in choosing. See a unit working, or don't pay much for it. You can't do any maintenance on a hard disc the way you can with a floppy drive. In my case, I had acquired a 10-megabyte Seagate half height drive. It is the same size as a half height floppy drive. I had expected to be able to run it from the floppy power supply, but the starting current load of 3.5 amps was too much, so I required a more powerful 'switched mode' type of power supply. Generally speaking, this type of supply is essential, and they are expensive, so budget for it. A new one might cost £80.

The Dragon interface board includes a controller chip, and data transfer takes place at two otherwise unused bytes above &HFOO. It is possible, therefore, with appropriate software, for the hard disc system to be used by Flex, OS-9 and native DragonDOS. However, Gordon only supplies OS-9 drivers. He feels, and I share this to an extent, that only under a sophisticated system such as OS-9 is it worth using a device such as a hard disc.

Set up

Having had the interface installed, and purchased the controller card and power supply, I was anxious to get up and running. The first thing to do is format the hard disc. A utility disc of OS-9 modules is provided. Because of the way the BOOT command works in DragonDOS, it is unfortunately necessary to BOOT from floppy, so that was what I did. The message 'OS-9 version 2.4 Welcome to Gordon's OS-9 system' greeted me.

> *Having seen friends and colleagues using hard disc systems that provide megabytes of fast online storage I thought, well, the answer to OS-9 problems. . .*

Using MDIR from my standard system disc showed the presence of the device descriptor/driver pair HDISK and HO, in addition to DDISK and DO/1/etc. The 'E' option showed HDISK's size as approximately equivalent to the DDISK module.

It was then necessary to format the hard disc. The FORMAT module on the utility disc is totally different from the standard system module. It asks for information about the hard disc — number of surfaces (heads) and sectors. This information must

be known, as incorrect data could in fact damage the hard disc.

A further piece of information required is the so-called interleave factor. This refers to the fact that physically consecutive sectors may not have consecutive numbers assigned by the system. This is due to the fact that in the time taken by the system to load and read a sector, the spinning disc has moved on. If the system had to wait until the physically consecutive sector was back in position, time would have been wasted. More efficient to read the sector that is under the head at the time. Setting the interleave factor obviously requires detailed knowledge about the speedof operation of the system. This is not available, so I took the default values, making a mental note to try different interleave factors in subsequent format operations. Then the operator must choose logical or physical format. The former is equivalent to format without verify, whereas the latter will verify, and lock out any unusable tracks.

The importance of a hard disc is proportional to its fragility. . .since backing-up is time consuming, I ended up not doing it, and lost a lot of work (serves me right).

Naturally on first format, the physical option is selected. Although the formatting process works much faster than on a floppy system, with so many sectors to format a considerable period of time is taken. The operator must stay with the system during this time in case any action need be taken. After an apparently interminable period of time (15 minutes) the disc is formatted, and the command FREE/HO came back with the satisfying answer '44,000 sectors'. The next step is to transfer all OS-9 programs to the hard disc.

Hierarchy

Bearing in mind the hierarchial directory system, preliminary thought should be given to the data structure. Essentially, one may choose to have one master CMDS directory full of commands, probably ordered into a number of subdirectories, or set up a directory for each application (e.g. one for Stylo, one for Pascal, etc) each with its own CMDS subdirectory.

This latter approach involves some duplication of files, but may be easier to use. Consider a directory called STYLO. Using the DSAVE command, the contents of the floppy distribution disc of *Stylograph* are transferred to the hard disc directory STYLO. When using the STYLO program, a file can be constructed with the BUILD command, or a text editor, to COPY over from the master CMDS directory to the STYLO CMDS subdirectory any additional commands that may be required. Others prefer to have the disc structure organised around a file's purpose rather than language (eg to work on household accounts files, CHX to a directory of HOUSEHOLD. ACCOUNTS rather than DYNACALC/ HOUSEHOLD.ACCOUNTS).

I chose to construct a master CMDS directory from the system disc, and a number of directories representing each floppy disc that I subsequently stored on the hard disc (by DSAVE) — one for BASICO9, one for STYLO etc. My OS-9 Group new discs I saved hierarchically under a directory of OS9NEWSDISKS.

Remembering that a pathlist is limited to 254 characters (keyboard buffer length), and to reduce opportunity for errors, short directory names are to be preferred. Another point to bear in mind is that only one file with a given filename can exist within a directory, so if DSAVE encounters a module with a given name in a directory, a revised updated module being loaded in from floppy will not be stored, DSAVE will report an error 218 and carry on. The user must then manually rename the existing or updated module and load the module from floppy.

In use

Because of the device independent I/O structure of OS-9, all commands that report the state of a device can be redirected to HO such as DIR, FREE, DCHECK. The first point that became apparent was that many applications had been hard-coded to refer to DO for various reasons. STYLO looks for the proportional spacing table on DO, SPELL looks for the dictionaries on DO, etc. There are two approaches to dealing with this either rename the HDISK driver to DDISK, and HO to DO. This will cause problems for floppy disc operation. Or use DEBUG or other utility to search for references to DO in programs, and patch to HO. Because of the size of HDISK, OS9Gen should be used to create a BOOT file of a minimum number of modules. Once all applications are patched to replace referenced to DO with HO, it is not necessary to refer to the floppy disc at all. Therefore, DDISK can be left out of the OS9BOOT, and loaded into memory from the startup file (thereby being able to be removed when not required). Having created directories for all my OS-9 applications, including separate directories for V2.OS9 and COCO.OS9, I have used approximately one quarter of the space in my 10 megabyte hard disc. This is with substantial duplication of files.

A major advantage is speed in use. There are two aspects to this — the data transfer rate of the hard disc, and the effective track access time are faster with a hard disc than with a floppy, so modules are loaded into memory, and files are saved from memory, and much faster with a hard disc than with a floppy.

Disc snags

Secondly, with all commands and applications on-line and accessible via CHX, there is no time spent swapping discs. For example, in V2 of OS-9 I can BOOT up, load a previously built file to call DEBUG and customise STYLO and the floppy device descriptors, load STYLO and construct a C program, save and compile it, go back to STYLO, construct a BASICO9 program which calls the C module.

The speed at which modules are loaded into memory means that the storage device is almost like an extension to ram. This is how the 64K limitation can be overcome. Regular use of OS9 gave me a different perspective on my home computer. My Dragon became a terminal to an abstract OS-9 computer. Now with the hard disc, the OS-9 computer takes on a more definite form. Since the hard disc system, by adding so much on-line storage, gives me the opportunity to personalise my OS-9 system, the peripheral device takes on a crucial importance.

There are snags, of course. The importance of the hard disc to my system is also proportional to its fragility. I've been running the system for approximately one year now. But I've never written about it before because I have suffered problems with the hard disc 'going down'. Since backing up is a difficult and time consuming problem, I ended up not doing it and when the system crashed, lost a lot of work (serves me right).

(Backing up hard discs is even more trying than backing up floppies and the consequences potentially more disasterous owing to the large amount of work capable of being stored. However, the principle that the files which you lose are always the hardest to restore is equally true of both types, says your Editor with feeling.)

I have carried out some benchmark tests on the performance of the HDISK/HO/10meg seagate combination, and they were a mix of respectable results (compared with current machines) and some poor results. Gordon has rewritten the driver so I intend to wait and report on the performance of the new version.

A number of people, users of other computers, have looked askance when I have described the system. Why, they say, would you want to bolt a powerful peripheral onto a rather humble machine like a Dragon? This displays a lack of understanding of the power of OS-9, and the different approach to computing given by the combination of powerful operating system and hard disc.

The total cost of about £150 — £200 may seem a lot of money, but if you are into OS-9, have a significant investment in software and a pioneering spirit, you should go for it.

Dragon Invoicer

Craig Henderson *plans a prosperous future.*

IF you have a printer to go with your Dragon, have you ever wondered how much more you could use it and improve on the presentation of your work?

As lately I have been keeping a much closer eye on my income and outgoings (credit and debit, getting into a good habit for when I leave school), I have written this invoice compiler to compile, print-out, edit, save and load all appropriate information. You can use it to keep a careful watch on your money.

When you first execute the invoice compiler, the title screen will be displayed and you will be asked if you would like to loan an existing invoice off cassette. Press (Y) or (N) correspondingly. If your reply is yes (Y), the program will then go on to explain what to do step-by-step and when the invoice is loaded, you will go into the main menu. If no (N), you will go straight to the main menu

With the main menu you will have five choices: pressing (1) will send the current invoice to the printer, as shown in **figure one**. The printer is set up for an 80 column printer, but this can be changed by poking location 155 in line 6 to the printer width and location 153 to location 155 divided by 4 (type POKE 153, PEEK(155)/4) and changing line numbers 2060/70/71 to contain the appropriate width minus 21. Pressing (2) will take you into the ENTER INVOICE mode. In this mode you will be asked to enter all the appropriate information to build up an invoice.

The screen will clear and the question "HOW MANY DAY ENTRIES?" will appear. Day entries refer to the lines of text in the invoice which have the status:

DAY MONTH DATE AMOUNT

The prompt for a title is next and then is followed by a prompt for the month(s) of work, then the date that you began work followed by the date that you retired from work. When you have done this, the screen will clear again and the title of the invoice will be displayed at the top of the screen, followed by the entry number and the prompt ENTER DAY. This is the day of the entry number which is displayed. If you don't enter a valid day, the machine will tell you so. When you have done this, you will be asked to enter the month, and again if you enter an invalid month, you will be told. Next enter the date and the amount. On the amount, if you just enter the figure, a hash (£) will be displayed in front of it later on, but if you precede the figure by a hash (£) yourself, that is OK.

Once you have done this for all the entries, you will see a display as in **figure two**.

You fill this in using keys 0-9 for the figures and the left and right arrow keys for editing. If you need to enter a 2-digit

```
PROGRAM NOTES.

5            -   CLEAR STRING SPACE
6            -   SET PRINT WIDTH
20-40        -   DATA FOR DAYS OF THE WEEK
50-60        -   DATA FOR MONTHS OF THE YEAR
90-140       -   PRINT TITLE PAGE
150-280      -   MAIN MENU AND SELECTION ROUTINE
300-350      -   QUIT ROUTINE.  NEW PROG.  BOOT MACHINE
1000-1130    -   LOAD INVOICE OFF CASSETTE
2000-2200    -   PRINT INVOICE TO PRINTER ROUTINE
2500-2980    -   ENTER INVOICE FROM KEYBOARD ROUTINE
3000-3110    -   SAVE INVOICE TO CASSETTE ROUTINE
3500-3820    -   EDIT INVOICE ROUTINE
4000-4070    -   LEFT ARROW MOVE ROUTINE FOR  EDITING  DATA
                 LINES

VARIABLES.

D            -   NUMBER OF ENTRIES
H            -   NUMBER OF DATA LINES
M$           -   MONTH(S) OF WORK
B$           -   BEGAN DATE
R$           -   RETIRED DATE
D$(1-D)      -   DAY OF CURRENT ENTRY
MN$(1-D)     -   MONTH OF CURRENT ENTRY
DT$(1-D)     -   DATE OF CURRENT ENTRY
A$(1-D)      -   AMOUNT ON CURRENT ENTRY
NW(1-H)      -   NUMBER OF WEEKS IN CURRENT DATA LINE
AW(1-H)      -   AMOUNT PER WEEK IN CURRENT DATA LINE
TW(1-H)      -   TOTAL FOR WEEKS IN CURRENT DATA LINE
AM$(1-H)     -   ADDITION INFORMATION

          <  TW(n)=NW(n)*AW(n)  >
```

```
PAPER ROUND INVOICE
DURING THE MONTH(S) OF JUNE AND JULY 1986
                         BEGAN  : WEDNESDAY JUNE 4TH
                         RETIRED : WEDNESDAY JULY 30TH

WEDNESDAY           JUNE            4TH          £2.20
WEDNESDAY           JUNE           11TH          £2.20
WEDNESDAY           JUNE           18TH          £2.20
WEDNESDAY           JUNE           25TH          £2.20
WEDNESDAY           JULY            2ND          £2.60
WEDNESDAY           JULY            9TH          £2.60
WEDNESDAY           JULY           16TH          £2.60
WEDNESDAY           JULY           23RD          £2.60
WEDNESDAY           JULY           30TH          £3.00

04 WEEKS/DAYS  @  02.20 EACH   =  £008.80
04 WEEKS/DAYS  @  02.60 EACH   =  £010.40
01 WEEKS/DAYS  @  03.00 EACH   =  £003.00
                       TOTAL   =  £022.20

SIGNED .............................................
```

```
DATA LINE EXAMPLE.

        --  WEEKS/DAYS   @    #--.-- EACH
        ^
            =  #---.---  TOTAL


DATA LINE ( FULL ) EXAMPLE


            05   WEEKS/DAYS   @   #02.20 EACH

            =  #011.00  TOTAL

IS THIS INFORMATION CORRECT ?
```

number in a 3-digit space or a 1-digit number in a 2-digit space, then start the number with a '0'. If the pence column is to be left blank, it is still a good idea to fill it with zeros (0).

When you have finished, another prompt: IS THIS INFORMATION CORRECT? will appear. Answer (Y) or (N) correspondingly. If (N) then you will go back to edit it and if (Y) you will proceed to the next prompt:

DO YOU WISH TO ENTER ANOTHER -- WEEKS/DAYS @ £--.-- EACH = £---.-- TOTAL?

Again answer (Y) or (N) correspondingly. If (Y) then you will go back with the same procedure as before and if (N) you will go on to be able to enter an extra line of notes, etc.

Press (ENTER) if no notes are required and you will return to the main menu. (If you have an existing invoice in memory which you don't want and you would like to enter another invoice, do not press (5) in the main menu as this will re-boot the machine, but press (BREAK) and then (RUN) and the memory will clear).

Pressing (3) enters the save invoice mode. This saves the current invoice in memory to the cassette recorder. It saves it as a file as you can see from the program, so be sure to insert the 'remote' plug to the cassette recorder. You are able to position the cassette with the remote plug inserted, because I have made use of the MOTOR and AUDIO commands within the software.

Pressing (4) enters the edit invoice routine. Each piece of information will then, in turn, be displayed followed by a question mark and a flashing cursor. If this piece of information is correct then press (ENTER) to pass on to the next. If this is not correct then enter the corrected text and (ENTER).

Pressing (5) will exit the program, but when you press (5) you will be asked if you are sure. If you are, then follow by (Y) and the program will end, display the boot-up message and the Basic program will be lost along with all the data which was in memory.

If you accidentally exit the program, (by pressing (BREAK) OR (RESET)) you can continue without losing the current invoice in memory by typing 'GOTO 10'. **NB** If you own a Dragon 64, check the rom routine addresses BEFORE using them.

Although this is a Basic program, you may not relish the task of typing it in, and therefore I will be happy to supply a copy on cassette for £2.00 including post and packing or for £1.30 if you supply a C15 or C20 cassette. Send a cheque or postal order made payable to Craig Henderson to Invoice Software, Craig Henderson, Sanray, 39 Woodbury Avenue, Wells, Somerset BA5 2XW. Anyone who sends too little money will have the cheque returned without any software. Happy Invoicing.

```
5 CLEAR 11000
6 POKE 155,80:POKE 153,20:POKE 328,0
10 ' ** INVOICE COMPILER **
20 DATA MONDAY,TUESDAY,WEDNESDAY
30 DATA THURSDAY,FRIDAY,SATURDAY
40 DATA SUNDAY
50 DATA JANUARY,FEBRUARY,MARCH
60 DATA APRIL,MAY,JUNE,JULY
70 DATA AUGUST,SEPTEMBER,OCTOBER
80 DATA NOVEMBER,DECEMBER
90 CLS
100 PRINT"************************************                  ***
*    INVOICE COMPILER          ****                    *************
*********************"
110 PRINT:PRINT"WOULD YOU LIKE TO LOAD AN     INVOICE OFF CASSETTE ?"
120 I$=INKEY$:IF I$=""THEN 120
130 IF I$="Y" THEN 1000
140 IF I$<>"N" THEN 120
150 CLS
160 PRINT@75,"MAIN MENU"
170 PRINT@107,"==== ===="
190 PRINT@199,"1) PRINT INVOICE"
200 PRINT@231,"2) ENTER IOVOICE"
210 PRINT@263,"3) SAVE INVOICE"
215 PRINT@295,"4) EDIT INVOICE"
220 PRINT@327,"5) QUIT PROG."
230 PRINT@448," SELECT : ";
240 I$=INKEY$:IF I$="" THEN 240
250 IF I$<"1" OR I$>"5" THEN 240
251 IF D=0 AND I$="4" THEN GOSUB 2020:GOTO 150
260 I=VAL(I$):PRINT I:SOUND 30,1:CLS
261 IF D<>0 AND I$="2" THEN 160
270 ON I GOSUB 2000,2500,3000,3500,300
271 Z$=INKEY$
280 GOTO 150
```

```
300 PRINT@70,"ARE YOU SURE [Y/N]"
310 I$=INKEY$:IF I$=""THEN 310
320 IF I$="N" THEN RETURN
331 IF I$<>"Y" THEN 310
340 CLS
350 EXEC 46080
1000   LOAD
1010 CLS
1020 PRINT"PLEASE PLACE CASSETTE INTO THE  CASSETTE RECORDER,INSERT THE
 'REMOTE' PLUG,CHECK VOLUME        SETTING AND PRESS A KEY.":AUDIOON:MOTORON:E
XEC 34091
1030 MOTOROFF:AUDIOOFF
1040 CLS
1050 PRINT@224,"PLEASE ENTER THE FILENAME FOR   THE INVOICE WHICH IS TO BE
     LOADED ";:LINE INPUT FL$
1060 CLS:PRINT@234,"SEARCHING"
1070 OPEN "I",£-1,FL$:PRINT@234," LOADING "
1080 INPUT£-1,D,H,B$,R$,M$,T$,AM$
1081 DIM D$(D),MN$(D),DT$(D),A$(D),NW(H),AW(H),TW(H),NW$(H),AW$(H),TW$(H)
1090 FOR OP=0 TO D-1
1100 INPUT£-1,D$(OP),MN$(OP),DT$(OP),A$(OP)
1110 NEXT:FOR OP=1 TO H
1120 INPUT£-1,NW(OP),AW(OP),TW(OP),NW$(OP),AW$(OP),TW$(OP):NEXT
1130 CLOSE £-1:GOTO 150
2000 ' PRINT INVOICE
2010 PRINT@226,"PLEASE MAKE SURE PRINTER IS    <ONLINE> AND PRESS A KEY":
EXEC 34091
2020 IF D=0 THEN CLS:PRINT@224,"NO INVOICE CURRENTLY IN MEMORY !":SOUND 100
,20:RETURN
2030 Z$=RIGHT$(T$,7):IF Z$="INVOICE" THEN PRINT£-2,T$:GOTO 2050
2040 PRINT£-2,T$;" INVOICE"
2050 PRINT£-2:PRINT£-2,"DURING THE MONTH(S) OF ";
2060 IF LEN(M$)<=59 THEN PRINT£-2,M$:GOTO 2080
2070 Z$=RIGHT$(M$,LEN(M$)-59)
2071 M$=LEFT$(M$,59)
2072 IF RIGHT$(M$,1)<>" " AND RIGHT$(M$,1)<>"," THEN 2074
2073 PRINT£-2,M$:PRINT£-2,Z$:GOTO 2080
2074 Z$=RIGHT$(M$,1)+Z$:M$=LEFT$(M$,LEN(M$)-1):GOTO 2072
2080 PRINT£-2:Z$="BEGAN    : "+B$
2082 Q$="RETIRED : "+R$
2090 OP=INT((PEEK(155)-LEN(Q$))/2)
2100 PRINT£-2,TAB(OP);Z$
2110 PRINT£-2,TAB(OP);Q$:PRINT£-2
2120 FOR OP=1 TO D-1
2130 PRINT£-2,D$(OP);STRING$((INT(PEEK(155)/4))-LEN(D$(OP))," ");
2131 PRINT£-2,MN$(OP);STRING$((INT(PEEK(155)/4))-LEN(MN$(OP))," ");:IF LEN(
DT$(OP))<4 THEN DT$(OP)=" "+DT$(OP)
2132 PRINT£-2,DT$(OP);STRING$((INT(PEEK(155)/4))-LEN(MN$(OP))," ");
2133 PRINT£-2,A$(OP):NEXT
2134 PRINT£-2:PRINT£-2
2140 FOR OP=1 TO H
2150 PRINT£-2,NW$(OP);" WEEKS/DAYS  @  ";AW$(OP);" EACH   = £";TW$(OP):NEXT
2160 LM=0:FOR OP=1 TO H:LM=LM+TW(OP):NEXT
2170 PRINT£-2,STRING$(23," ");"TOTAL   = £";
2171 BN$=STR$(LM):BN$=RIGHT$(BN$,LEN(BN$)-1)
2174 IF LM<10 THEN BN$="00"+BN$:PRINT£-2,BN$;:GOTO 2180
2175 IF LM<100 THEN BN$="0"+BN$:PRINT£-2,BN$;:GOTO 2180
2180 IF LM=INT(LM) THEN PRINT£-2,".00" ELSE IF LEN(BN$)<6 THEN PRINT£-2,"0"
 ELSE PRINT£-2
2190 FOR OP=1 TO 10:PRINT£-2:NEXT
2192 PRINT£-2:PRINT£-2:PRINT£-2,"SIGNED ";STRING$(PEEK(155)-7,".")
2200 RETURN
2499 END
2500 ' ENTER INVOICE
2510 INPUT"HOW MANY DAY ENTRIES ";D
2520 IF D>356 THEN PRINT"MAXIMUM IS 356 - TRY AGAIN":GOTO 2511
```

```
2530 DIM D$(D),MN$(D),DT$(D),A$(D)
2540 N=D:D=1
2550 PRINT:LINE INPUT"PLEASE ENTER THE TITLE OF THIS  INVOICE ";T$
2555 PRINT:LINE INPUT"PLEASE ENTER THE MONTH(S) OF   THIS WORK ";M$
2556 PRINT:LINE INPUT"PLEASE ENTER THE DAY/MONTH/DATE WHICH YOU BEGAN WORK
 ";B$
2557 PRINT:LINE INPUT"PLEASE ENTER THE DAY/MONTH/DATE WHICH YOU RETIRED FRO
M WORK  --> ";R$
2560 CLS:PRINT T$
2570 PRINT:PRINT"ENTRY NUMBER ";D
2580 LINE INPUT"ENTER DAY ";D$(D)
2590 H=0
2600 READ E$:IF E$=D$(D) THEN H=1
2605 IF H=0 AND E$="SUNDAY" THEN PRINT"INVALID DAY":RESTORE:GOTO 2580
2610 IF H=0 THEN 2600
2620 RESTORE
2630 LINE INPUT"ENTER MONTH ";MN$(D)
2640 H=0
2650 FOR O=1 TO 7:READE$:NEXT:E$=""
2660 READ E$:IF E$=MN$(D) THEN H=1
2670 IF H=0 AND E$="DECEMBER" THEN PRINT"INVALID MONTH":RESTORE:GOTO 2630
2680 IF H=0 THEN 2660
2690 RESTORE
2700 LINE INPUT"ENTER DATE ";DT$(D)
2710 LINE INPUT"ENTER AMOUNT ";A$(D)
2720 IF LEFT$(A$(D),1)<>"£" THEN A$(D)="£"+A$(D)
2721 IF INSTR(A$(D),".")=0 THEN A$(D)=A$(D)+".00":GOTO 2730
2722 IF INSTR(A$(D),".")>LEN(A$(D))-2 THEN A$(D)=A$(D)+"0":GOTO 2722
2730 D=D+1
2740 IF D<=N THEN 2560
2741 H=1
2750 CLS
2760 PRINT"-- WEEKS/DAYS  @  £--.-- EACH":PRINT:PRINT"= £---.-- TOTAL"
2770 P=32:PRINT@P,"^";
2780 I$=INKEY$:IF I$=""THEN 2780
2781 IF I$=CHR$(8) THEN 4000
2782 IF I$=CHR$(9) THEN 2800
2790 IF I$<"0" OR I$>"9" THEN 2780
2800 PRINT@P-32,I$;:PRINT@P," ";:P=P+1
2810 IF P=64 THEN P=96
2811 IF P=34 THEN P=51
2812 IF P=53 THEN P=54
2813 IF P=56 THEN P=99
2814 IF P=102 THEN P=103
2815 IF P=105 THEN 2830
2819 PRINT@P,"^";
2820 GOTO 2780
2830 PRINT@256,"IS THIS INFORMATION CORRECT ?"
2840 I$=INKEY$:IF I$=""THEN 2840
2850 IF I$="Y" THEN 2900
2860 IF I$<>"N" THEN 2840
2870 PRINT@256,"":GOTO 2770
2900 PRINT@256,""
2901 L=1024
2910 FOR OP=112 TO 121
2920 IF PEEK(L)=OP THEN NW$(H)=NW$(H)+CHR$(OP-64)
2930 NEXT:NW(H)=VAL(NW$(H))
2940 IF L=1024 THEN L=L+1:GOTO 2910
2950 L=1043
2951 FOR OQ=112 TO 121
2952 IF PEEK(L)=OQ THEN AW$(H)=AW$(H)+CHR$(OQ-64)
2953 NEXT:AW(H)=VAL(AW$(H))
2954 IF L=1043 THENL=1044:GOTO 2951
2955 IF L=1044 THEN AW$(H)=AW$(H)+".":L=1046:GOTO 2951
2956 IF L=1046 THEN L=1047:GOTO2951
2957 L=1091
```

```
2958 FOR OP=112 TO 121
2959 IF PEEK(L)=OP THEN TW$(H)=TW$(H)+CHR$(OP-64)
2961 NEXT:TW(H)=VAL(TW$(H))
2961 IF L=1091 THEN L=L+1:GOTO 2958
2962 IF L=1092 THEN L=L+1:GOTO 2958
2963 IF L=1093 THEN L=L+2:TW$(H)=TW$(H)+".":GOTO 2958
2965 IF L=1095 THEN L=L+1:GOTO 2958
2965 IF H<11 THEN PRINT@256,"DO YOU WISH TO ENTER ANOTHER    -- WEEKS/DAYS
 @  £--.-- EACH    = £---.-- TOTAL ?" ELSE GOTO 2969
2966 I$=INKEY$:IF I$=""THEN 2966
2967 IF I$="Y" THEN H=H+1:GOTO 2750
2968 IF I$<>"N" THEN 2966
2969 CLS:PRINT"-- WEEKS/DAYS  @  £--.-- EACH    = £---.-- TOTAL":PRINT:PRINT
"ENTER ANY EXTRA LINE IF WISHED  (<ENTER> TO END - REMEMBER THE  PRINTER HA
S 80 CHARACTERS ACROSSSO BEWARE OF LAYOUT !)"
2970 PRINT:PRINT:PRINT:LINE INPUT AM$
2980 RETURN
3000 ' SAVE INVOICE
3005 IF D=0 THEN FOR OP=1 TO 100:NEXT:GOTO 2020
3010 PRINT"PLEASE PLACE THE CASSETTE INTO  THE CASSETTE RECORDER,SET IT TO
<RECORD> MODE,CHECK THE VOLUME  SETTING,INSERT 'REMOTE' PLUG ANDPRESS A KEY
":AUDIOON:MOTORON:EXEC 34091
3020 CLS:MOTOROFF:AUDIOOFF
3030 CLS:PRINT@224,"PLEASE ENTER FILENAME TO BE     SAVED ";
3040 LINE INPUT FL$
3050 OPEN "O",£-1,FL$
3060 PRINT£-1,D,H,B$,R$,M$,T$,AM$
3070 FOR OP=0 TO D-1
3080 PRINT£-1,D$(OP),MN$(OP),DT$(OP),A$(OP)
3090 NEXT:FOR OP=1 TO H
3100 PRINT£-1,NW(OP),AW(OP),TW(OP),NW$(OP),AW$(OP),TW$(OP):NEXT
3110 CLOSE £-1:RETURN
3500 ' EDIT INVOICE
3510 PRINT"TITLE : ";T$
3520 INPUT Q$:IF Q$="" THEN 3540
3530 T$=Q$
3540 PRINT"MONTHS : ";M$
3550 INPUT Q$:IF Q$="" THEN 3570
3560 M$=P$
3570 PRINT"BEGAN : ";B$
3580 INPUT Q$:IF Q$="" THEN 3600
3590 B$=Q$
3600 PRINT"RETIRED : ";R$
3610 INPUT Q$:IF Q$="" THEN 3630
3620 R$=Q$
3630 FOR OP=1 TO D-1
3640 PRINT D$(OP):INPUT Q$:IF Q$="" THEN 3660
3650 D$(OP)=Q$
3660 PRINT MN$(OP):INPUT Q$:IF Q$="" THEN 3680
3670 MN$(OP)=Q$
3680 PRINT DT$(OP):INPUT Q$:IF Q$="" THEN 3700
3690 DT$(OP)=Q$
3700 PRINT A$(OP):INPUT Q$:IF Q$="" THEN 3720
3710 A$(OP)=Q$
3720 NEXT
3800 PRINT AM$:INPUT Q$:IF Q$="" THEN 3820
3810 AM$=Q$
3820 RETURN
3999 STOP
4000 PRINT@P," ";:P=P-1
4010 IF P=102 THEN P=101
4020 IF P=98 THEN P=55
4030 IF P=53 THEN P=52
4040 IF P=50 THEN P=33
4050 IF P=31 THEN P=32
4069 PRINT@P,"^";
4070 GOTO 2780
```

Dragonsoft

New software for review should be sent to *Dragon User*, 49 Alexandra Road, Hounslow, Middx TW3 4HP.

Continued from page 11
that they have laser beams that cannot be seen. This prevents the enemy from dodging the lasers, and makes your ships almost totally indestructable.

However, as part of Utopia's defences, there is a fleet of kamikaze shuttle craft which, instead of firing at you, will home in on your ship and attempt to collide with it, unless you can stop them.

The planet surface is strewn with objects. Many of them are defences, some are just plain obstacles to avoid. Among the defences are the aforementioned shuttle craft, rocket launchers and force fields.

To aid your progress in the game, there are various fuel pods strewn around the planet which must be flown over in order to raise your fuel levels.

It must be noted that the main aim of this game is not to achieve outstanding scores, although it is nice to do so, but to retain enough fuel to enable you to fly out over the planet.

A score is achieved each time you destroy an enemy craft, although your score can be considerably increased when your ships are flown over certain strips of land.

Should your fuel fall below zero or should you hit one of the numerous objects, you will immediately die and be sent back to the beginning of the game.

This is a shortcoming of many games; *Rola Ball* suffers from the same problem. Why programmers continue to use this irritating routine puzzles me. The net effect is that the user will become very bored with the game if, every time he or she dies, they are sent back to the beginning to repeat what has just gone before. It is particularly annoying when you are just a short distance from the end of the game — not that I have got that far with this one.

Utopia is a difficult game, perhaps too difficult. Maybe some of you will say that this provides a challenge, but the game really does suffer from only giving you three lives. Nobody can be expect to complete a massive game like this with only three lives. No doubt the hackers will come up with a miracle cheat code which will dramatically increase lives.

The graphics are really first class, I would say they're among the best I have seen, although the screen and your ship do occasionally flicker as the screen scrolls. The sound and music is excellent; the title music in particular is pleasant and well put together.

Utopia is a bit like a former game for the BBC called *Fortress. Utopia* doesn't look anything like *Fortress*, but the aim is virtually the same and the two games have the same feel — and like *Fortress*, it is all too easy to get killed!

For my money *Utopia* is one of the best Dragon games of all time despite minor faults, and Jonathan Cartwright and the rest of the crew at Starship deserve praise for the work on it.

Donald Morrison



No red herrings

Program: *The Great Fish Van Scandal*
Supplier: Orange Software
Price: £2.99

THE Great Fish Van Scandal is one of several adventures from the newly created Orange Software. If ever an adventure had a ridiculous storyline behind it, this game certainly does.

One day a strange thing happens to you: as you walk to work, you are jumped by two masked men, blindfolded, tied up and gagged and thrown in the back of what smells like a fish van. Then you are hit on the head, and when you regain your senses, you are in an even smellier damp cell. (*Alternatively, negotiating the London Underground during the Thursday night rush after a brief spell of socialising, you unaccountably find yourself in Euston Station loo without remembering how or even why you got there — One Of Our Correspondents*).

It sounds as though somebody has a pretty overactive imagination! The game starts off in the damp cell, with water dripping from the ceiling and rats scurrying beneath a pile of straw.

The only visible object is a blanket. EXAMINE BLANKET reveals that there is nothing special about the blanket. With no exits apparent, the only direction is east to another part of the cell, where there is a grille in the ceiling. The only way out would be to but your way out, but as you have nothing sharp at the moment, you'll have to find another way.

Bakc to the first locations, and I soon realised that the only way out was under that pile of straw, though quite how I still don't know.

The game supports the usual verb/noun inputs with NSEWUD for directions. Verbs include look, examine, help, score, but, kill etc. and the usual load/save commands.

However, despite the inclusion of verbs like kill, the game will not tolerate violence — on receiving a violet command, the game will stop and the cocmputer will cold-start. Don't panic — it's not a bug. As soon as you press a key the game will restart with a warning not to use violence.

There are a large number of locations in the game, and many of them contain elements of humour, though the game hovers precariously between humour and dodgy language in parts. (*Mr. Morrison offered an example of a dodgy abbreviation here, but as I couldn't work it out, I will leave it to greater wits than myself. — Ed.*)

The game operates totally in text, which hasn't been redefined. Nevertheless, the author has mananged to keep his descriptions fairly detailed and the screen is quite well laid out.

Although the game operates in Basic, it plays at a reasonable speed. As with most Basic games now, it's well protected with the Break key being disabled as well as the LIST command.

There seems to be a slight bug in the game, as every time you type EAT you bite your lip and eventually die, regardless of whether or not you are eating anything at the time.

It's a good, well thought out game with a totally original idea behind it. Not one perhaps for the beginner, as its fairly complex, but should appeal to the hardened adventurer.

Donald Morrison



Communications

Problem: Can anyone supply me with a copy of the instructions for using the Gemini *Database* program? Costs reimbursed.
Name: Mark Matthews
Address: 14 Grantley Close, Ashford, Kent TN23 1UE.

Problem: Help, Grosvenor/DMK RTTY cartridge will not load. S/H appears or cursor disappears and keyboard goes dead. Where has the company gone?
Name: P. Edwards
Address: 41 Queensway, Soham, Ely, Cambridgeshire CB7 5BU. 0353-722466.

Problem: Wanted: any football pool programs.
Name: Peter Marsh
Address: 37 Cobbetts Walk, Bisley, Surrey GU24 9DU.

Adventure trail

Adventure: *The Cricklewood Incident/Ten Little Indians*
Problem: How do you travel by bus or train/Just HELP!
Name: Neil Davies
Address: 4 Grinshill Drive, Crowmoot, Shewsbury, SY2 5JE.

Raise your graphs

*Plots by **R L Moss**, lines by the Dragon*

THE following program will create polar and parametric cartesian graphs ofthe kind often encountered in A-level maths syllabi. The program as it stands assumes the use of single disc drive and program giving text on the graphics screen, such as *Hi-res* or *Rainbow Writer*, but neither is essential. A screen dump of the graph can be made in conjunction with a screen dump program, of which there are seemingly thousands available, for your particular printer.

Type in the program as listed with the following changes:

For cassette users, delete LOAD and RUN commands from the first line. Screen dump and hi-res programs must be loaded in separately (if used). Change LOAD and RUN commands in line 0 to that of names of the screen dump and text-reformatter program (eg. Hi-res, RW64, etc). Screen dump should be in mid memory to avoid clashing with the reformatter or Basic program. Delete if not used. If a text-reformatter program is not used, delete line 1 and change line 2 to:

```
2CLS:FORN=OTO12:READA$:?A$:?"
";:NEXT
```

Change line 200 to:

```
200CLS:SCREEN1,0:I$=INKEY$:IFI$="
"THEN200
```

If a Basic screen dump is used, type this in from line 500 onwards ending with a RETURN statement. Change line 340 to:

```
340 GOSUB500:GOTO200
```

Otherwise line 340 should contain the EXEC-address of the screen-dump program.

Once RUN the program displays the graph of the function contained in line 10. For a polar graph, line 10 should contain a DEF FNR(0)=f(0) where f(O) is the function to be drawn, and the letter O is the variable used. This should be followed by POLAR=1, and an example is included in the listing.

Cartesian graphs are displayed using two equations, DEF FNX(T)=f(T) and DEF FNY(T)=g(T). For an example see below. For non-parametric equasions of the form y=f(x) use

DEF FNY(X)=f(X): DEF FNX(X)=X.

The variable POLAR should be set to zero.

Line 11 contains respectively the y- and x-shifts to move the axes on the screen, upper and lower limits for variables, the size of the plot and the gap between each point, all of which need experimenting with. A larger value for GAP gives a smoother trace, but seems to take for ever. All these can be changed when the program is running. After drawing the graph, the computer awaits a one key command;

S will change the size of the plot
G changes the gap between the dots
X and Y will change x- and y-shifts respectively
O will save graph to higher memory — useful for overlaying two graphs.
I will recall stored graph, and draw current graph over it, but make sure axes are the same.
L or K alters limits. L give limits in pi.
D or P will print graph to printer providing screen dump loaded.
SPACE will draw a fully lined graph (no dots)
SPACE a second time will give a plot of all radii of curvature for polar graphs.
ENTER gives a quick summary of all keys.

Examples to try:

10 DEF FNY(T)=SIN(T)*SIN(T)*SIN(T) :DEF FNX(T)=COS(T)*COS(T) :POLAR=0 11 YS=0:XS=0:UL=PI*2 :LL=0:SI=100:GA=.5

10 DEF FNY(X)=SIN(X):DEF FNX(X) =X:POLAR=0

11 YS=0:XS=0:UL=PI:LL=-PI :SI=60:GA=-.1

10 DEF FNR(0)=SIN(3*0)^3 +1:P0=1

11 YS=0:XS=0:LL=0:UL=PI*2 :SI=10:GA=.25

10 DEF FNR(0)=SIN(3*O)^2 +1:POLAR=1

11 YS=0:XS=0:UL=0:UL=PI*2:SI=50 :GA=.5

```
0 PCLEAR8:CLEAR200,20000:PMODE4:COLOR0,1:PCLS:SCREEN1,0:LOAD"DUMP.BIN",20001:RUN
"RW64.BIN"
1 IFPEEK(1030)=45THEN5:ELSEFORN=1024TO1536:POKEN,128:NEXT:FORN=0TO12:READA$:FORM
=1TOLEN(A$):P=1024+N*32+M+3:IFN=0 ORN=12 THENP=P-3
2 POKEP,ASC(MID$(A$,M,1)):NEXTM,N
5 PI=ATN(1)*4
10 DEF FNR(O)=SIN(O)+1:POLAR=1
11 YSHIFT=-50:XSHIFT=0:UL=PI*2:LL=0:SIZE=70:GAP=1
12 IFPOLAR THEN DEF FNY(O)=SIN(O)*FNR(O):DEF FNX(O)=COS(O)*FNR(O)
20 IFUL=PI/2 ANDLL=0 ANDPOLAR=1 THEN LINE(0,0)-(0,191),PSET:LINE-(255,191),PSET:
YY=191:XX=0:ELSEYY=96-YS:XX=128+XS:LINE(0,YY)-(255,YY),PSET:LINE(XX,0)-(XX,191),
PSET
30 IFRO=1 THENLINE(XX+SI*FNX(LL),YY-SI*FNY(LL))-(XX+SI*FNX(LL),YY-SI*FNY(LL)),PS
ET
40 FORO=LL TOUL STEPPI/(GAP*100)
50 Y=FNY(O):X=FNX(O)
60 Y=YY-Y*SI:X=XX+X*SI
70 IFSGN(Y)=-1 ORSGN(X)=-1 THENNEXT:GOTO200
80 IFROUND=0 THENPSET(X,Y,0)
90 IFROUND=1 THENLINE-(X,Y),PSET
100 IFROUND=2 THENLINE(XX,YY)-(X,Y),PSET
110 NEXT
200 PRINT@0,"";:I$=INKEY$:IFI$=""THEN200
210 ONINSTR(1," SLKXYGCIODPH"+CHR$(13)+CHR$(12),I$) GOTO220,250,260,265,270,280,
290,300,320,330,340,340,360,360,300:GOTO230
220 RO=RO+1:IFRO=3THENRO=0
```

Colour connection

Ray Deakins makes a hardware mod to balance the Dragon's colour.

THE December issue of *Dragon Answers* contained a letter asking if it is possible to correct the colour balance on a D32. The following short article might just help. I am assuming that the writer of the letter has already tried adjusting the UHF modulator to no effect.

The D64 (and some of the later 32s) use a small variable capacitor instead of the fixed value 15pf capacitor (C7 on the 32 — C32 on the 64). The value of this should be between 2 and 22 picofarads. (Maplin part. no. WL70M.) The component is located on the main circuit board top left hand side, immediately below the crystal XL1. IF you are handy with a soldering iron it should not take very long to make this modification. With the new capacitor in place and the computer switched on adjust the small screw with a non-magnetic, insulated screwdriver until you get the best colour response.

Changing this capacitor has improved the colour display on the two Dragons that I have modified in this way (both with issue 4 power boards and the Astect UM1287 modulator). The issue 3 power boards have the UM1286 modulator and a slightly different circuit. Although I see no reason why this should not work, I cannot guarantee that this will cure the problem, and suggest that anybody contemplating making this modifications should only go ahead if they are fully confident in their ability to do the job, and reverse it if necssary!

The following short program will help to set up the colours. To use the program to its full advantage, it is essential that your TV is set up correctly, ie colour balance, etc. Display time can be altered by changing 1500 to any desired value.

The National Dragon Users Group, or Peaksoft, may be able to help with circuit diagrams if required.

Incidentally, 1988 is the year of the Dragon! Let's keep it that way.

```
10  CLS0:PRINT"BLACK"
20  FOR N=0 TO 1500:NEXT N
30  CLS1:PRINT"GREEN"
40  FOR N=0 TO 1500:NEXT N
50  CLS2:PRINT"YELLOW"
60  FOR N=0 TO 1500:NEXT N
70  CLS3:PRINT"BLUE"
80  FOR N=0 to 1500:NEXT N
90  CLS4:PRINT"RED"
100 FOR N=0 to 1500:NEXT N
110 CLS5:PRINT"BUFF"
120 FOR N=0 TO 1500:NEXT N
130 CLS6:PRINT"CYAN"
140 FOR N=0 TO 1500:NEXT N
150 CLS7:PRINT"MAGENTA"
160 FOR N=0 TO 1500:NEXT N
170 CLS8:PRINT"ORANGE"
180 FOR N=0 TO 1500:NEXT N
190 GOTO 10
```

Listing one cont.d

```
230 IFNOT(CL)THENPCLS
240 GOTO20
250 Q=SI:GOSUB450:INPUT"SIZE ";SI:GOTO400
260 GOSUB460:PRINT"NEW LIMITS x PI ";:INPUTLL,UL:LL=LL*PI:UL=UL*PI:GOTO400
265 PI=1:GOSUB460:PRINT"NEW LIMITS ";:INPUTLL,UL:PI=4*ATN(1):GOTO400
270 Q=XS:GOSUB450:INPUT"X-SHIFT ";XS:GOTO400
280 Q=YS:GOSUB450:INPUT"Y-SHIFT ";YS:GOTO400
290 Q=GA:GOSUB450:INPUT"NEW DOT GAP ";GA:GOTO400
300 PRINT"CLS ";:CL=NOT(CL):IFCL THENPRINT"OFF"ELSEPRINT"ON"
310 WAIT1000:GOTO200
320 PCOPY5TO1:PCOPY6TO2:PCOPY7TO3:PCOPY8TO4:GOTO20
330 PCOPY1TO5:PCOPY2TO6:PCOPY3TO7:PCOPY4TO8:GOTO200
340 EXEC20001:GOTO200
360 SCREEN0,0:IFINKEY$="" THEN360 ELSESCREEN1,0:GOTO200
400 I$=INKEY$:PRINT@0,STRING$(10,32):PRINTSTRING$(15,32):GOTO200
450 PRINT"CURRENT=";Q:RETURN
460 PRINT"LOWER";LL/PI;"UPPER";UL/PI:RETURN
1000 DATA SPACE- DIFFERENT PATTERN,S - CHANGE SIZE,G - NEW DOT GAP,L - CHANGE LI
MITS IN PI,K - CHANGE LIMITS,X - X-SHIFT   + GOES RIGHT,Y - Y-SHIFT   + GOES UP,
C - CLS ON/OFF,I - LOAD IN STORED GRAPH,O - STORE GRAPH,H - HELP SCREEN
1010 DATA D - DUMP TO PRINTER,ANY OTHER KEY REDRAWS
```

Winners and Losers

Every month Gordon Lee will look at some prize programming

THE announcement of the winning entries from the February competition in May's *Dragon User* must have sent shock-waves of curiosity through the computing public. "How was it done?", I hear you ask between gasps of amazement. 'It' being the computation of the one hundred digit product of two fifty digit numbers in a time of under 10 seconds! In fact, the winning entry achieved a time of under 4.3 seconds — reduced from the 278 seconds running time of the example given in the question.

The timings given are those found by averaging a number of trial runs. The second decimal place may vary from run to run (or from machine to machine), but these timings should be typical from the listings given here. Note that the use of any 'speed' poke is not permitted and that the setting of the timer to zero and the printout of the time taken *must* be the first and last command of your program respectively.

The method used by the three runners up was basically similar. This was to reduce the number of calculations to be made by splitting the two fifty-digit numbers into groups of 5 and 4 digits respectively, and then multiplying eachof these groups in turn, the product being inserted in the answer string at the correct position. In this way the first fifty-digit number can be split into ten five-digit sections, while the second number can be split into thirteen four-digit sections. In this way the multiplication part of the computation can be done in just 130 steps. The Dragon can handle the multiplication of a five and four digit number into a nine-digit product without any difficulty and the running time can be significantly reduced.

The winner, Pete Faraday, was able to do even better and complete the calculation in half the time of his nearest rival. Once again, this was achieved by taking groups of digits but the edge was gained in converting these to an array of *numeric* variables. As Pete says in his note of explanation:

"A$ is broken down and stored in the array A. B$ is similarly broken down, and each part is multiplied by the stored parts of A$. These products are added into an array Z. Only after this is done are the carries within Z calculated and the values within Z corrected. Z is converted back to a string."

One interesting feature of three of the listings is the use of a command which, although acceptable on the Dragon, is not always possible on other machines. As far as I am aware, it is also a command which is not found in the manual supplied with the Dragon, although it is quite a useful command. Readers are probably quite familiar with the command

Z$ = MID$(A$,4,3)

which places in variable Z$ the three characters from A$, beginning at 4.

Not so well known is the 'inversion' of this command which is shown in use at line 20 of the following short listing:

```
10 A$ = "ABCDEFGHIJKL" :Z$ = "XYZ"
20 MID$(A$,4,3) = Z$
30 PRINT A$
```

Quite predictably, this produces the not unexpected result of 'ABCXYZGHIJKL", and yet this is a command that is not accepted by some versions of Basic.

D J Gray (8.98 seconds)

```
1 TIMER=0:CLEAR 10000
2 A$="73795304344252726633390147358974818114755211845351"
3 B$="30658548775849541883672359302221775860324499365539"
4 IF (LEN(A$)/5-INT(LEN(A$)/5))<>0 THEN A$="0"+A$:GOTO 4
5 IF (LEN(B$)/4-INT(LEN(B$)/4))<>0 THEN B$="0"+B$:GOTO 5
6 Z$=STRING$(LEN(A$)+LEN(B$),"0")
7 FOR I=LEN(A$)-4 TO 1 STEP-5:A=VAL(MID$(A$,I,5)):S=0
8 FOR J=LEN(B$)-3 TO 1 STEP-4
9 S=VAL(MID$(Z$,I+J+4,4))+A*VAL(MID$(B$,J,4))+INT(S/10000)
10 MID$(Z$,I+J+4,4)=RIGHT$(STR$(S),4)
11 NEXT J
12 C$=MID$(STR$(INT(S/10000)),2):MID$(Z$,I+(5-LEN(C$)),LEN(C$))=C$
13 NEXT I
14 IF LEFT$(Z$,1)="0" THEN Z$=MID$(Z$,2):GOTO 14
15 PRINT Z$
16 PRINT"TIME TAKEN ";TIMER/50;"SECONDS"
```

R H Wilson (9.86 seconds)

```
100 TIMER=0:CLEAR300
110 A$="73795304344252726633390147358974818114755211845351"
120 B$="0030658548775849541883672359302221775860324499365539"
130 A=LEN(A$):B=LEN(B$):C$=STRING$(A+B,48)
140 FOR C=A TO 1 STEP-5:D=VAL(MID$(A$,C-4,5)):H=0:M=B+C-3
150 FOR E=B TO 1 STEP-4:F=VAL(MID$(B$,E-3,4))
160 G=D*F+VAL(H$)+VAL(MID$(C$,M,4)):G$=STR$(G)
170 MID$(C$,M,4)=RIGHT$(G$,4):H$=LEFT$(G$,LEN(G$)-4)
180 M=M-4:NEXT E:H=VAL(H$)+VAL(MID$(C$,M,4)):H$=STR$(H)
190 H$="0000"+MID$(H$,2):MID$(C$,M,4)=RIGHT$(H$,4):H$="":NEXTC:
200 X=0:IF LEFT$(C$,1)="0"THEN C$=MID$(C$,2):X=1
210 IF X=1 THEN 200
220 PRINT C$:PRINT"TIME TAKEN ";TIMER/50
```

Pete Faraday (4.26 seconds)

```
10 TIMER=0:CLEAR 500
20 A$="73795304344252726633390147358974818114755211845351"
30 B$="30658548775849541883672359302221775860324499365539"
40 AL=LEN(A$):AU=INT((AL-1)/4):T$=STRING$(4*AU+4-AL,"0")+A$:DIM A(AU)
50 AL=AU*4+4:FOR AX=0 TO AU:A(AX)=VAL(MID$(T$,AL-3-4*AX,4)):NEXT
60 BL=LEN(B$):BU=INT((BL-1)/4):T$=STRING$(4*BU+4-BL,"0")+B$:DIM Z(AU+BU+1)
70 BL=BU*4+4:FOR BX=0 TO BU:B=VAL(MID$(T$,BL-3-4*BX,4)):FOR AX=0 TO AU
80 Z(AX+BX)=Z(AX+BX)+A(AX)*B:NEXT:NEXT:Z$="":R=10000:C=0
90 FOR ZX=0 TO AU+BU:Z=Z(ZX)+C:C=INT(Z/R):Z$=MID$(STR$(Z+R-C*R),3)+Z$:NEXT
100 Z$=MID$(STR$(C),2)+Z$
110 IF ASC(Z$)=48 THEN Z$=MID$(Z$,2):GOTO 110
120 PRINT Z$
130 PRINT"TIME TAKEN:";TIMER/50;"seconds"
```

S A Siddiqui (4.42 seconds)

```
10 TIMER=0:PCLEAR1:CLEAR28700:C=0:Z=C:V=C:F=5:P=C:Z$="":N=C:O=C:A=C:R=.00001:T=1
00000:N$="0000":S=99999:D=-1:M=C:K=C
20 A$="73795304344252726633390147358974818114755211845351"
30 B$="30658548775849541883672359302221775860324499365539"
40 B$="00000000"+B$:L=LEN(B$)-4:DIM B(L*.2):K=LEN(A$):A$="000"+A$:Z$=STRING$(K+
50 FOR Z=L TO 1 STEP-F:D=D+1:B(D)=VAL(MID$(B$,Z,F)):NEXT:L=L-4
50 FOR M=K TO 1 STEP-4:O=L+M
70 A=VAL(MID$(A$,M,4))
80 FOR N=0 TO D:P=O-F*N
90 V=B(N)*A+C:C=INT(R*V)
100 Z=VAL(MID$(Z$,P,F))+V-C*T
110 IF Z>S THEN C=C+1
120 MID$(Z$,P)=RIGHT$(N$+MID$(STR$(Z),2),F)
130 NEXT:NEXT
140 PRINT STR$(VAL(LEFT$(Z$,8)));MID$(Z$,9):PRINT TIMER*.02;"SECONDS ELAPSED"
```

Write: ADVENTURE

Pete Gerrard rings changes on theme

I received an interesting letter recently, from one Robert Troughton in Keighley (Yorkshire, wrong side of the Pennines, but we'll let it pass), who has this to say: "I've been trying to write my own adventure recently, but haven't much idea what to do or how to go about doing it". He's developed a fairly reasonably parser (quoting "Take the large sword and drop it" as an example), but then ends up with a general plea for help and ideas.

Since it occurred to me that a number of people might be in a similar position to Robert, that of knowing the basics (and possibly Basics) but not being quite sure how to go about putting it all together, we'll spend this month trying to help you all along a little.

Choose your system

Most people who write adventures either end up developing a system of their own and then fitting everything into that, or they use an adventure writing system developed by someone else and have to first of all modify their own way of thinking so that the adventure they're contemplating will be able to be written by using that particular system. It's no use coming up with an idea that allows you to swap from one character to another if the system in use has no way of coping with that.

So first of all ensure that the game you're going to be writing can actually be written! Especially if you're going to be submitting a design rather than the finished thing. Study other games by that company, and make sure that what you're sending them will fit in with what they have already done. Otherwise you'll get a rejection along the lines of "great idea, but our system won't be able to handle it". Translated, this means that they probably like the idea but are not willing to spend the time and energy in extending their system to cope with whatever vagaries you've introduced. And why should they? For every adventure that won't fit into their way of programming they've probably got half a dozen ideas that will.

For a simple and totally unoriginal example of this, take the age-old problem of having to find a key to open a locked door. Having found the key, the player might reasonably be expected to go back to the door and type in something like OPEN DOOR. A simple line of code like:

```
IF key carried AND correct location AND
door locked THEN unlock door
```

will sort everything out. Most adventure systems can be expected to cope with something like that, although I'll admit to a personal preference for not particularly liking games that have seemingly hundreds of doors and hundreds of keys scattered about the place, forcing you to spend an eternity finding the right one for the right door. Then, when you do, and type in OPEN DOOR, the program responds with THE DOOR IS LOCKED. I know it is, stupid, that's why I've spent half an hour finding the key and am now telling you to open the wretched thing!

From little problems mighty adventures grow, and if you are going to have a locked door then for heaven's sake put something interesting behind it so that the player gets some slight feeling of reward after opening it. Perhaps make it a double, or even a triple, barrelled problem. The lock might be covered with a panel that is rusty with age and cannot move, and when you do find some oil and free it you then discover that there is something stuck inside the barrel of the lock and you have to drag it out. A little extension from an ancient problem makes all the difference to the player.



Too many adventures, though, seem to rely on locking and unlocking things in order to succeed. Doors, chests, cabinets, safes, most of the games that I've seen need something to be unlocked at some time or another, so another of the 'getting you started' ideas might well be to leave out locks altogether. It might make the game harder to write, but it will be a touch more original.

And since originality seems to be the name of the game these days, leave out mazes as well. Easy enough to program though they may be, I personally find them tedious in the extreme:

```
IF P(CP,NO)<>0 THEN CP=P(CP,NO):
GOTO 10
```

Or something similar, is all that is needed to get a player moving around, in or outside of a maze, if the array P(x,y) contains the number of locations and the number of available exits, CP is the current location, and NO is the direction that the player wants to go in. However, moving around within a maze could easily be achieved by simply having six or so locations that all have the same location description, such as "You are in a maze of twisting tunnels, all alike" to quote the original. Then, whenever the player moves, the location description remains the same and the player has no idea whether the move has been advantageous or not. This sort of maze is relatively easy to solve by dropping objects and mapping your progress like that: if the object's still there when you move, then you're in the same location. On the other hand, the programmer can make things more difficult for the player by having something like this in the DROP routine:

```
IF (CP<46 AND CP>53) THEN OB(NO)=
0:PRINT message
```

This assumes that the maze is in locations 47 to 52, and the array OB holds the current location of all objects. Setting that location to zero would mean that the object had vanished, and the message could be something along the lines of "As you drop the (object) a genie appears in a puff of smoke and steals it, saying that he'll leave it outside the maze for safe keeping. He vanishes in a cloud of greasy smoke". Thus the player can now no longer map the maze by dropping objects.

A logical maze

Whether or not you have a maze is, of course, up to you. I don't like them myself, but some people do actually seem to enjoy solving them. But if you do have a maze, try and give it a purpose. So many of them just seem to be there for the sake of being there and being a nuisance, with no reason behind this sudden appearance of a maze in the middle of nowhere. Make it a constructive maze (a London adventure might use Hampton Court, for instance) with something at the centre of it, perhaps. Players do like to feel that they've achieved something by solving a maze.

Another old chestnut, which can be given a variation on a theme in order to get the budding adventure game writer started, is the perennial problem of light and dark. Many, many games have you roaming around looking for a torch before you can go into a particular set of locations, otherwise you won't be able to see where you're going in the dark and fall into a pit to be later consumed by a monstrous Grue.

```
IF (CP>32 AND CP<40) AND PD=1 THEN
PRINT message
```

That sort of line could easily be added to the movement routine, with locations 33 to

39 being the dark ones. If the variable PD is set to 1 then the player has not found a light source, and the message could read something like this: "As you try to move in the dark you trip and stumble, almost knocking yourself out as you fall into a black pit. The last thing you see before you faint from sheer terror are the burning eyes and slavering jaws of some unknown beast". Peter waxes lyrical!

Some adventures do allow you to roam about in the dark without falling down anywhere, but of course you cannot see objects that might be hidden on the ground. This is wrong, everyone's eyes adapt to the dark eventually, so a solution might be to make the player stay in the dark locations for a set number of moves before they can see the glittering emerald lying half-concealed behind a rock. Or, you may care to consider having a light source that goes out after a certain number of moves.

Then again, you might care to have something totally different. One adventure of mine featured a cave area that had to be explored, and there was also a torch. Unfortunately the torch didn't work, and never would work, so some other alternative had to be found. This alternative took the form of a guide dog, who would happily guide you through the caves and prevent your feet from stumbling, in return for a spot of food. Added to this was the length of time taken in the caves, so that the player's eyes adjusted slightly and were able to make out a set of tennis rackets in one corner of a particular cave. Tennis rackets? It was a strange adventure!

A troll's pint

One final point to consider for the beginner to the world is another favourite that could easily be changed in order to add character to the game. This is where a certain being, such as a troll, blocks your path and will not let you past until you give him something:

```
IF CP=45 AND OB(troll)=CP THEN
PRINT message
```

If the troll is blocking your path, and it's location 45, then print a message to the effect that "The troll sneers at you from the other side of the rickety old bridge, and refuses to let you pass". You might try burning your bridges in front of you, only to be told "The troll doesn't like people playing with his bridge, and stamps out the fire before it can take hold".

Almost inevitably, the item that you have to give the troll will be one that you want to use again later, so that the player will have to think of some way of getting it back afterwards. But you could do this same sort of problem in a totally different way. You could invite the troll for a drink or something, and when he gets to the tavern and studies the door to see if any other trolls have been in he could find that there were no runes at the inn and so go elsewhere to find a drink, leaving you plenty of time to scurry across the bridge. If it was a rope bridge you could roll it up after you and thus prevent the troll getting back across afterwards. All variations on a theme, and with that, goodbye!



Professor Deadrock time again, after last month's whinge against people who should know better, and as promised we have the old boy's rather unique solution to an adventure that I get a number of enquiries about, namely *Pyramid of Doom*. Thanks to Michael Edwards and R. Stedham for giving the good Professor a helping hand on those occasions when his feet stumbled slightly.

Dear Diary,

Pyramids and sarcophagi (or is that sarcophagusses? Makes them sound like some sort of aquatic mammal, a herd of sarcophagusses approaching the starboard bow, Cap'n) abound on my latest travels abroad, and one cannot help but wonder what Binder would have made of it all. Still, ours is not to reason why, and having left the admirable courier behind (decent enough egg, but does try rather too hard to ensure that we old fogies have what he refers to as a 'good time') I had the misfortune to stumble slightly and had to grab a pole to keep my balance. Heaven preserve us, the pole was in fact a shovel, and fancying that such an implement might come in useful late on I took hold of the thing and went to a nearby pool. A curious key and some liquid of a rather dubious nature from the canteen soon came into my possession, and after travelling first east, then north, then east again, I leant on the shovel to pause for breath and found myself digging!

I unearthed a somewhat small key, and after taking it went south, whereupon I was digging with gusto, as that familiar Deadrock instinct told me that something of value might well be revealed hereabouts. A large hole was revealed, to be precise, and upon going into it I found a door. Soon unlocked that, I can tell you, and immediately retreated up again. There I dropped the shovel and took hold of a stone. Something about it caught my fancy, can't say what. I unlocked the main door and went in, feeling vaguely guilty as I did so, for no apparent reason. Perhaps it is that fear, inherent in most of us, of being caught where one ought not to be. What would the courier say? In a fit of excitement I dropped both keys and lit my flashlight. I was in the Pyramid of Doom, and immediately took a rest before pressing on further. It would not do to be caught unawares.

I chanced upon a sarcophagus (wretched thing) and opened it. Proceeding by some inner instinct I headed north then east, and lo and behold I found a flute. Strange looking artefact, I can tell you. A quick jaunt west, south, and south once more, and I was in an unusual place. A little foible of mine, I know, but in such circumstances I could not help but try and relax by playing the flute. Not that I am one of nature's natural tunesmiths, but there you have it.

I studied the fireplace and found some coal and ashes, although I had to look twice before the ashes revealed themselves. I made a mental note to check with my optician after the holiday was concluded. The ashes looked strangely un-natural, and upon further examination I was able to find a necklace! That could come in useful at the next bridge meeting. Having got the necklace I entered a passage and immediately headed north and east. Some strange graffiti could be found here (I use the word graffiti, hieroglyphics become so tiresome after a while) and upon reading it I decided to drop the stone and necklace. The stone was becoming something of an unwanted burden anyway. A little jerky attracted my attention (curious word, and one cannot help but fear for the future of the English language), and I took it before proceeding further.

I went west, then north, where I saw an astonishing sight. An oyster was looking straight at me! In my alarm at this unexpected intrusion on my privacy I dropped the jerky, whereupon the oyster immediately ate it and presented me with a pearl. Naturally enough I took a keen hold of the pearl as I trotted off towards an archway that sadly failed to resemble that glorious example of architecture at Marble Arch. On going east I unaccountably dropped the pearl, and in my haste to find it I hurried west and north. Somehow I also managed to lose track of the flute, but I was never much of a virtuoso and James Galway need have no fear of my musical talents displacing him.

Red as coal

I proceeded west and south and returned to the wretched sarcophagus. Some inner instinct seems to constantly drag me back towards it, I cannot think what it all might mean. I went down, losing my step slightly as I did so, and chanced to brush my arm against the coal that I was carrying and thus cleaned it. It was a ruby, but clearly not one of any great value and certainly not something that would be worth taking down the Portobello Road for evaluation. What had caught my eye and caused me to be less than my usual careful self were some burning leaves, and I immediately thought of that blessed liquid from the canteen. I poured water all over them, extinguishing them, and was able to get what was obviously an immensely important tapestry. Peruvian, I shouldn't wonder, although I may be wrong. I am no great expert. I went to an alcove, and peered intently at a box that was lying there. I did a double take, looked again, and there was an iron glove. The sort of thing that Mike Tyson would wear, no doubt, although he would rue the day he crossed me I can assure you. Not that that is likely to happen, as I have a very forgiving nature. An old skull took my attention, and on giving it the most thorough of examinations I was able to ascertain that it had some gold teeth that were most definitely not National Health Service issue. I added them to my growing list of goods.

I performed my usual trick of heading west and south, but in my haste I somehow managed to catch my arm on something, a protrusion from the wall I fancy, and accidentally let the skull fall from my grasp. Good heavens above, what is the world coming to?! I hurried north as fast as I could possibly go, and still further I went in my efforts to escape. Fearing attack I donned the iron glove as a suitable source of weaponry, but my hands had become so drenched with perspiration in my flight that I initially dropped the blessed thing. With my patience rapidly approaching the end of its tether I replaced the glove and proceeded. I was going so fast that I almost collided with a door, and only a rapid motion of my hand prevented me from tumbling head-first against it. I daresay that to the casual observer if probably looked as if I were hitting the door, and so promptly removed the glove for fear of detection. I don't think that I will have any further use for it.

Gemmed to death

I found a rope, and my old training told me that this could be of immense use, so naturally enough I carried it with me. I headed south and south again, back to where I started my headlong flight, and went to the ladder. By the merest chance I stumbled slightly, and so was forced to go to the ladder a second time. I threw the rope up to the ceiling, to help me get up there, and had visions of the most awesome pharoah striking me dead for my impertinence. What could I possibly fling at this apparition that appeared before me? The ruby was worthless, and feeling somewhat akin to David against Goliath I threw the thing. Fortunately my aim was good and my blow was true, and I slew the fiend. I sneaked a sapphire into my possession as reward and offered myself a pat on the back. I did not turn the offer down.

I headed south and examined the strange looking collection of rubbish in front of me. An explorer! In my turn I examined him also, although this latter act was performed with some little distaste as I most certainly did not like the cut of his jib. For my pains I was rewarded with a pin and a carving, and after that loathsome task was concluded I went west and down, dropping the canteen as I did so but replacing its presence with a saw. My mother did not raise me to be a beast of burden but that, I fear, is what I am rapidly becoming as I am forced into being a human ship of the desert.

My journeying became ever more extravagant as first I went north, then up, then west, then north again. Some inner stirring of my heart was guiding me, of that I am sure. A rather crochety looking table seemed bent on being sawn into two smaller tables, and this I did. Upon completion of this task, and at the cost of an aching elbow to remind me that I am no longer the athlete of my youth, I dropped the saw and took hold of the necklace that had appeared. I stepped out smartly to the east and the somewhat ludicrous archway, where I could carry my baggage no more and so dropped tapestry, teeth, necklace, pin, carving, and sapphire. I felt ten years younger, and after a slight pause to catch my breath I trotted off towards the west and then turned north and west again.

Beetling home

I retrieved my saw, and after stepping out smartly to the south I retrieved both my keys, thanking the good Lord that no-one had appeared to steal them. Once more I returned to the wretched sarcophagus and went down, then south, then to the ladder. Once more I headed south and proceeded to my rope, following which I opened the chest and took out the most impressive looking crown that it has ever been my pleasure to behold. Feeling somewhat foolish, and rather as if I were in my own bathroom back home, I pulled the chain. I went to the stairs, sawed my way through the bars, and was finally able to drop the saw for the last time. Wearing the iron glove for safety I unlocked a coffer before dropping both keys, again for the last time. Inside the coffer was a bracelet, which naturally I took.

I went back to the window, got a bar by that curious inner feeling known only to us Deadrocks, and then performed the most extraordinary series of movements, so bizarre that I almost became giddy and can only set them down as I performed them: west, down, down, west, down, north, east. Such confusion! I went through the door, but unfortunately slipped and was forced to put out both hands to defend my face. On feeling the floor I discovered a coin, and took it with me as I headed east, lighting my flashlight in the gloom as I went. What I thought was a scurrying beetle turned out to be a stationary scarab, so I took it. West, then west again, re-lighting my possibly faulty flashlight as I walked, thereafter heading south, up, west, north and east before ending up once more at the ludicrous archway. East again, and added to my collection the scarab, coin, bar, crown, and bracelet. "A score of treasures" I murmured, but must have spoken louder than intended, for on saying the word "score" I was transported home and my latest holiday was complete.

What an interesting time he does have to be sure! See you next month.

Is this a record?

Gordon Lee really is going round in circles — almost

MOST of us are probably familiar with the old catch question 'How many grooves are there are a gramophone record?' The answer, of course, is two — one on each side. A similar problem asks 'How far does the stylus travel while playing one side of an LP record?'

Both of these depend on 'catches' in the way that the questions are put. However, a person who sees through the first of them is more likely to be trapped by the second. The 'trick' to both is dependent on the person seeing that one side of a record contains, to all intents and purposes, a single spiral groove which commences at the outer edge and then spirals in towards the centre. Consequently, when asked the second question, the temptation is to try to calculate the approximate length of this spiral track and to give this as the answer. In fact, the true distance that the stylus travels is somewhere in the region of four inches! — that is, the distance that it swings from the outer edge to the inner 'music' groove on the disc. The fact that the record is rotating beneath the stylus does not affect the distance of actual travel — except to guide it slowly towards the centre.

The spiral found on a typical gramophone record is of the type known as an Archimedian spiral. This is the locus of a point which rotates around a second point, the distance between those points decreasing at a constant rate. Consequently, the distance between adjacent coils of the spiral will remain constant. (Yes, I know that for certain technical reasons in the manufacture of records, this distance may vary slightly depending on the dynamic range of the music being recorded — but you get the idea!). A child on a playground roundabout who walks at a constant speed along one of the radial handrails would move in an Archimedian spiral in relation to the ground.

```
100 PMODE4:SCREEN1,0:PCLS
110 P=0.1:X=128:Y=92:R=90
120 FOR S=0 TO 1 STEP 0.02
130 E=S+0.02
140 CIRCLE(X,Y),R,1,1,S,E
150 R=R-P
160 IF R<0 THEN 160
170 NEXT:GOTO 120
```

A reasonable representation of an Archimedian spiral can be drawn on the Dragon's hi-res screen using either of the two listings given here. **Listing one** is a rather more straightforward method of using the computer's 'circle' command to draw small circle segments, each time reducing the radius a fraction to move the line in towards the centre. This is not a true spiral, but is instead a series of arcs of circles, but the end result is quite convincing. Interesting effects can also be obtained by drawing elliptical spirals. This is done by altering the 'height-width' ratio of the CIRCLE command at line 140. page 103 of the Dragon manual describes how this is done.

```
100 PMODE4:SCREEN1,0:PCLS
110 P=0.05:X=128:Y=92:R=80:
    PI=3.1416
120 S=SIN(2*PI/90)
130 C=COS(2*PI/90)
140 J=0:K=1
150 DRAW"BM208,92"
160 FOR A=0 TO 90
170 L=J*C+K*S:M=K*C-J*S
180 M$="M"+STR$(INT(X+R*M))
    +","+STR$(INT(Y+R*L))
190 DRAW M$
200 J=L:K=M:R=R-P
210 NEXT IF R>0 THEN 160
220 GOTO 220
```

A more finely-tuned spiral can be obtained using **listing two** which uses the sine and cosine functions to compute the x,y coordinates of each part of the spiral. In both of these listings, variable P is used to control the 'pitch' of the spiral and can be altered to adjust the rate at which the line converges on the centre.

Another type of spiral is that obtained by constructing the 'involute' of a circle. Imagine that you are confronted by a fierce dog who, fortunately is tethered to a tree by means of a long rope. By keeping just out of reach of the dog, and by moving round the tree, it would be possible to make the dog wind the rope around the tree. The path that it would take would be the 'involute' of the circle represented by a cross-section of the tree. Here, it is the tangent of the generating circle that is decreasing at a constant rate (as represented by the rope) rather than the radius as in the Archimedian spiral. To the naked eye, these spirals would both seem pretty similar, the difference being mainly in their methods on construction.

To complete the picture mention should be made of a third type of spiral, the logarithmic spiral. This is the most widely found spiral which occurs in nature, from the very small (the arrangement of the seeds in a daisy head) to the very large (the shape of many of the spiral galaxies in

Prize

Competitors who survive the giddy spirals of Gordon Lee's imagination will be wanting a rest. To assist them, we have negotiated a property deal with Dragon magnates **Preston Software**, last seen dealing in desirable island homes: ten copies of **Hotel On Mayfair**, the classic game inspired by *Monopoly*, await the weary winners. Take up your stations ...

Rules

Have you established a new record for the most effortless and most elegant way of measuring the distance from the plonk to the klunk on a LP record short of listening to the latest chart topper by Sade (ryhmes with g'day)? Or are you all a bunch of namby-pamby compact disc players? Pack your answer and program in a piece of paper marked AUGUST COMP, add any famous last words and post it to us. That hotel could be yours.

But the tiebreaker! Let us not forget the tiebreaker! Just complete the well known phrase or saying: 'If *I* had a hotel on Mayfair, I would ....every day.' Impress us with your hedonism. (Go and look it up in the dictionary.) (Heck. Who's got my dictionary?)

May winners

Well, we had a pretty miserable entry this month. This is what comes of letting it be known that Gordon Lee and the Editor are available for offers of marriage. We were only joking. Honestly, you can come out now, it's quite safe ... come out from under that table at once.

I shall sulk. However, I shall put it off till after the prizegiving, which goes like this: best correct answers, S A Siddiqui of Acton, T. Fawcett of Hendon, D J Gray of Middlesbrough, P. Morgan of Bristol, P. Weedon of Wotton-under-Edge, A J Westwood of Northampton, E A Newman of (Helen has lost the letter, but we know who you are, E A), J Smith of Twyford, Austan Henderson of Bromsgrove, and S. Beach of Chichester.

We have also created a special category for Denis O'Mulloy called "disqualified because he didn't actually enter the competition, but the rest of his letter was so good that we had to give him a mention." But seriously, Denis, no tiebreaker, OK, but no answer? Clearly your thoughts were on the Tournament of the Mind, for which many thanks.

Best tiebreaker, for sheer bad taste, was S A Siddiqui for "Tie a yellow Dragon round the Old Oak Tree."

All these persons will be receiving a copy of **Bean Stalker** from **Orange Software** in due course. Ignore the remark in last month's editorial. The editor was early for once.

Solution

See opposite.

outer space).

For the competition we are returning to our hypothetical gramophone record. A twelve-inch diameter long-playing record spins at thirty-three and a third revolutions per minute. The first 'music' groove is about a quarter inch from the outer edge and it ends just two and three quarter inches from the centre of the record, which plays for 23 minutes. Can you devise a simple program to compute the approximate length of the spiral groove on one side of the record?

The answer should be your assessment of the total length of the spiral.

The Answer

This is Gordon Lee's own solution to the May competition
see page 26 for results

THE square root of 2 (to 125 decimal places) is:

1.41421356237309504880268872420969
80785696718753769480731766797379 90
7324784621070388503875343276415727
350138462309122970249 2483

thus, the final five digits are '92483' (as the next digit in the computation is a 6 a value of '92484' would be acceptable.

This value was computed using a refined method of trials, starting with the value 1414 as the root land 19999396 as its square. The decimal place was dispensed with during the operation and replaced at the end. Each successive digit was determined by trial, starting with a 9 and decreasing by 1 each time. The required value was the first that was found which took the square to a value with a left hand digit of 1, For example, the square of 14143 is 200024449, but dropping the final digit of the root to a 2 we find that 14142 when squared gives 199996164. Therefore, this will determine the correct value. This procedure is continued as often as required in order to bring the square as close to, but not exceeding, 1.9 recurring (1.999999...)

The actual method was performed using string variables to hold the digits a method outlined in the past in *Dragon User*. To eliminate the need for the whole calculation to be performed from scratch, a method of computation was devised to allow the revised result to be appended to the previously computed total. The method used is outlined below, although a full explanation of how it works is too lengthy to be explained here.

1) Take the current value of the square and multiply by 100
2) Take its root as previously computed and multiply by 20. Then multiply by the digit being tested.
3) Compute the square of the digit being tested.
4) Add these three values together.
5) This gives the revised square.

For example, we have found the square 1999396 and its root 1414. We need to find the next digit for the root. So, starting with 9 we repeat the procedure outlines above until the resulting square falls below 19999999, the number of nines depending on the stage of the calculation. In this case, the digit will be 2.

So, taking each step as above, we get:
1) 1999396 x 100 = 1999396000 (actually done by putting 00 at the end of the string holding this value).
2) 1414 x 20 x 2 = 56560
3) 2 squared = 4
4) Add these together = 19996164
This gives the revised square 199996164 and its root 14142. So the process is repeated until the required degree of accuracy is obtained. As the square will contain twice the number of digits as its root, the maximum length of a string (255 characters) will be adequate for the computation to be made without having to 'split' the digits into more than one string.

If the square root value given is tested using the multiplication routine given as **listing one** on page 26 of the February 1988 *Dragon User*, its square can be shown to be 1999999 recurring. Altering the final digit (3) to a 4 will produce 200000 recurring, ie a value which is too high.

Communication

Write down your problem on the coupon below (make it as brief and legible as possible) together with your name and address and send it to Communication, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP.

Problem . . . .

Name . . . .
Address . . . .

Adventure Contact

To help puzzled adventurers further, here is an Adventure Helpline — simply fill in the coupon below, stating the name of the adventure, your problem and your name and address, and send it to Dragon User Adventure Helpline, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP.

Adventure . . . .
Problem . . . .

Name . . . .
Address . . . .

CLASSIFIED ADS

HERE'S MY CLASSIFIED AD.
(please write your copy in capitals on the lines below)

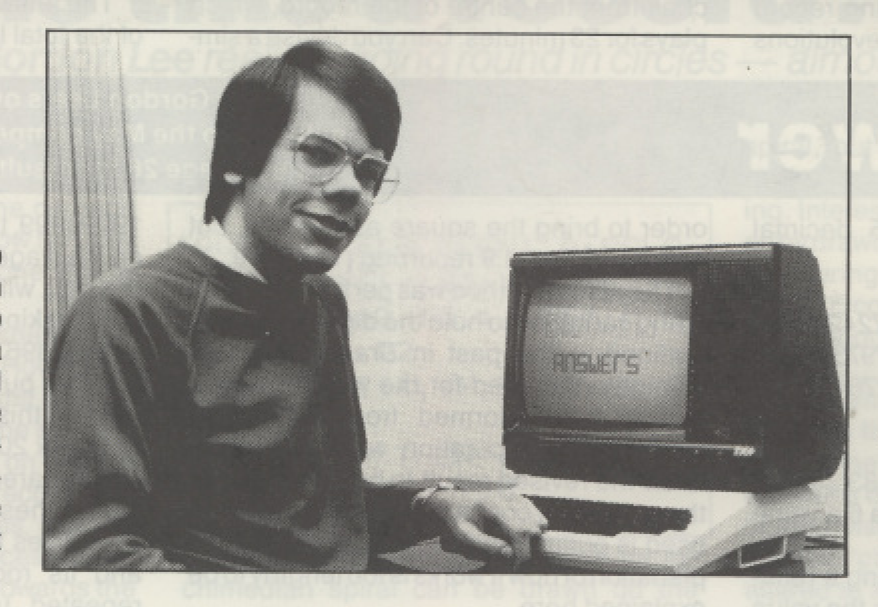
Name . . . .
Address . . . .
. . . . **Tel:** . . . .

35p per word. Addresses must be paid if required, including post code. Please send this form to Classified Dept., Dragon Publications, 49 Alexandra Road, Hounslow, Middlesex TW3 4HP.

Dragon Answers

If you've got a technical question write to Brian Cadge. Please do not send a SAE as Brian cannot guarantee to answer individual inquiries.

Making a line tick



I would like to know if it is possible to draw a line of an certain length at an angle to another line. I would like to be able to draw the second hand of a watch ticking round. The only way I have found of doing this is to use DATA statements to hold co-ordinates of the lines to be drawn — this is very time consuming as the co-ordinates have to be worked out on graph paper for each new circle. Is there a better way?

Andrew Lamb
2 Marsham Road
Hazel Grove
Stockport

A little elementary geometry is all that is needed here. Using the SIN and COS functions of Dragon Basic (which work in radians, not degrees), the following program will draw a second hand centred on the point XC,YC and of length R, ticking round the 60 second positions.

See page 145 of the Dragon Basic manual for an explanation of these functions. I'll leave it to you to add hour and minute hands to the clock!

```
10 PMODE 4,1:PCLS:SCREEN 1,1
20 R=60:XC=128:YC=96
30 CIRCLE(XC,YC),R+5
40 FOR S=0 TO 59:REM SECONDS OF A MINUTE
50 X=XC+R*COS((S/60)*6.28-1.57)
60 Y=YC+R*SIN((S/60)*6.28-1.57)
70 LINE(XC,YC)-(X,Y),PSET
80 TIMER=0
90 IF TIMER<45 THEN 90
100 LINE(XC,YC)-(X,Y),PRESET
110 NEXT S
120 GOTO 40
```

Eprom makes sense?

I bought an eprom programmer from Compusense at the last London show. I need some more information on how to save a program from memory to eprom, how do I access the program?

Some cartridge cases have two sockets with one or two eproms, others only have one, does this make a difference?

Peter Gent
77 Maulsham
Chelmsford
Essex CM2 9PY

I assume that you have instructions on how to actually use the eprom programmer (if not then get them from Compusense). You may be able to put Basic programs into eprom and run them there if you carefully set up the variable table pointers and Basic program pointers correctly (see the *Firmware* series for details), but you are more likely to be successful with machine code programs which need to be ORGed at 49152 to run in an eprom.

Most cartridges do have two sockets, these take 4K eproms (making a total of 8K). There are two types of one socket cartridges, the most common contains a single 8K eprom; more rare are the ones that contain a single 16K eprom (the maximum).

Sound under machine code

IS it possible to use the SOUND command in machine code? If so, where do I store the note number and duration, and what address do I call?

TO use the Dragon's SOUND command from machine code programs, first store the pitch value at location 140 (this is the same as the first number in the Basic SOUND command). The duration should be stored at locations 141/142 as a 16-bit (msb first) number. This should be four times the duration given as the second argument in the Basic SOUND command.

Then immediately call the routine at 47781. This will produce the required sound. The value in 141/142 is decremented in the interrupt routine, so it is important to only set this up immediately before the JSR 47781.

Hi-res in two parts

I am planning to write a graphic adventure game, but I have a few problems. How can you (in machine code) split the hi-res screen in two, the upper half in semi-graphics 24 and the lower half in Pmode 4. I've seen this done in *The Dark Pit*.

Ola Eldoy
Stokken
N-5410 Sagvag
Norway

IT is possible to switch modes mid-screen. To do this you need to enable the horizontal sync interrupt (by using location $FF01), so that an IRQ occurs for every screen line, rather than at the end of each whole screen. Then using a counter to count the IRQs, you can switch modes at a particular scan line.

The drawback is that all these IRQs tend to slow down your main program somewhat. The method is not simple and not for the faint hearted, but it was well explained last year in a *Dragon User*.

More memory

PLEASE could you answer the following questions which may be of some interest to other readers. Can the Dragon 64 have more memory attached to it than 128K, and if so how, by cartridge or internal board? Can the Dragon have a second cpu connected and can it be a 68000 so that I can run Gem?

David Linsley
12 Glen Barr
South Pelaw
Chester-le-Street
Co. Durham
DH2 2JN

THE Dragon's built-in sam (synchronous address multiplexer) which controls the dynamic ram cannot handle more than 128K itself. However, there is no reason why extra ram should not be added either on board, or in cartridge, which would be bank switched with its own control circuits. However, I do not know of anyone producing add-ons to give in excess of 128K.

The Dragon cannot have a second cpu connected, and Gem needs a lot more than 128K to work with, anyway.